

Universidad de las Ciencias Informáticas
Facultad 7



**Sistema para la gestión de los proyectos educativos
de las brigadas docentes en la Facultad 7**

Trabajo de Diploma para optar por el título de
Ingeniero en Ciencias Informáticas

Autora: Dianela Castillo Mesa

Tutor: Ing. Yurién Ricardo Fuentes Guerra

Cotutora: Yaima Anido González

La Habana, junio 2011

“Año 53 de la Revolución”

Datos del contacto

Tutor: Ing. Yurién Ricardo Fuentes Guerra.

Ingeniero en Ciencias Informáticas, graduado en la Universidad de las Ciencias Informáticas (UCI) en el año 2008. Actualmente es Jefe de pruebas en el Grupo de Calidad del Centro de Informática Médica (CESIM) de la UCI. Es profesor de Ingeniería de Software en la propia universidad.

Cotutora: Ing. Yaima Anido González.

Graduada de Ingeniería en Ciencias Informáticas en la UCI (2008-2009), Recién graduada en adiestramiento. Pertenece a la Subdirección de Calidad del Centro de informática médica (CESIM).

*Agradezco a mi madre querida, porque ha sido la fuerza que me ha impulsado,
mi sostén en todo momento y la luz que ha guiado mi camino.*

*A mi padre por su comprensión y apoyo en los momentos difíciles de esta
etapa.*

*A mi hermano Manuel por ayudarme en cada etapa de estudio de mi vida y
por haber estado siempre ahí, cuando lo necesite.*

*A mi tía Silvita, a mi tío Frank y a mis abuelos queridos, por ser personas
tan especiales en mi vida y por quererme día a día sin importar mis defectos.*

A Aldi y Maideli por ayudarme a dar los primeros pasos en este trabajo.

*A mis amigos Lorenzo Barzaga Meriño y Mileydís Cruz, por ser mi apoyo
incondicional durante el desarrollo de mi tesis.*

*A mis queridas amigas y hermanas Lizandra y Yanicet, por haber estado ahí
en los momentos más difíciles de la carrera.*

A mis compañeros desde 1ro hasta 5to año, a quienes siempre recordaré.

A mis tutores por ayudarme en el desarrollo de la tesis.

A todos aquellos que de una forma u otra han formado parte de este sueño.

A todos, gracias.

Diane.

Dedico este trabajo de diploma en especial a mi mamá, por su amor incondicional, por dedicarme toda su vida, por cultivar mis valores, por ser mi ídolo y por ser la luz de mi mundo.

A mi abuelo Fructuoso Castillo porque aún sin estar conmigo, me ha guiado en mis sueños.

A mi padre y hermanos, a quienes adoro tanto.

A mis abuelos por estar conmigo aún estando lejos.

A mi tía Silvita por ser otra madre para mí.

Agradezco a mi tío Remi, a mi tía Idania, y a mis primos por ser personas tan especiales en mi vida.

A la madre de mis hermanos, a quien quiero tanto.

A mi nani Graciela por ser una persona tan dulce y con quien compartí gran parte de mi tiempo.

A mis amigos de la guagua que quiero mucho y que siempre recordaré.

A mis amigos Lizandra, Yanicet, Yilian, Lorenzo, Mileydi, Sergio y Alexei, a cada uno de ustedes gracias por regalarme un pedacito de su tiempo.

Diane.

Resumen

El presente trabajo surge del estudio realizado, a los proyectos educativos de la facultad 7 de la Universidad de las Ciencias Informáticas. La gestión de los mismos, en la facultad se realiza de forma semi-automatizada, lo que provoca diferentes dificultades, entre las que se encuentra, la pérdida de la información, a los estudiantes se les dificulta conocer la planificación de las actividades. Dada esta situación surgió la necesidad de crear un sistema para la gestión de los proyectos educativos de las brigadas docentes de la Facultad 7.

Para el desarrollo del sistema se utilizó el Proceso Unificado de Desarrollo (RUP) como metodología de desarrollo, como herramienta para el modelado del mismo se empleó Visual Paradigm en su versión 6.4. Se utilizó el framework de desarrollo el Symfony 1.3.3, con el lenguaje de programación PHP y como sistema de gestor de base de datos el MySQL. En el diseño de la interfaz se utilizó la librería Ext Js.

El sistema para la gestión de los proyectos educativos de las brigadas docentes de la facultad 7 (GEDUC). Posibilita agilizar el proceso que se lleva a cabo en los proyectos educativos, registrando toda la información. Manteniéndola visible para sus usuarios, evita que pueda ocurrir la pérdida de algún proyecto y facilita al coordinador de año así como al profesor guía el trabajo que realiza en los mismos. Este sistema es una aplicación amigable, fácil de usar que mejora, controla y viabiliza este importante proceso.

Introducción	1
Capítulo 1 Fundamentación Teórica	5
1.1 Concepto básico asociado al negocio	5
1.2 Estado del arte	5
1.2.1 Sistema automatizado Enter The Group	5
1.2.2 Propuesta de software	6
1.3 Tecnologías y Herramientas	7
1.3.1 Metodologías de desarrollo de software	7
1.3.1.1 Programación extrema (XP)	7
1.3.1.2 SCRUM	8
1.3.1.3 Proceso Unificado de Rational (RUP)	9
1.3.2 Lenguaje de modelado	10
1.3.2.1 Lenguaje de modelado OO i*	10
1.3.2.2 Lenguaje de modelado OO UML	11
1.3.3 Framework de desarrollo	12
1.3.3.1 Symfony	12
1.3.3.2 Kumbia	12
1.3.3.3 CakePHP	13
1.3.3.4 Comparación entre los framework estudiados[15]	13
1.3.4 Arquitectura	15
1.3.4.1 Patrón modelo vista controlador (MVC)	15
1.3.5 Lenguaje de programación para la Web	16
1.3.5.1 Lenguaje ASP.NET	16
1.3.5.2 Lenguaje PHP	16
1.3.5.3 Lenguaje Java Script	17
1.3.6 Herramientas case	17
1.3.6.1 Rational rose	18
1.3.6.2 Visual paradigm	18
1.3.7 Entorno de desarrollo	18
1.3.7.1 Net Beans	19
1.3.7.2 ZendStudio for Eclipse	19
1.3.8 Sistemas gestores de base de datos	19
1.3.8.1 MySQL	20
1.3.8.2 PostgreSQL	20
1.4 Propuesta de herramientas para el desarrollo	21
Capítulo 2: Características del Sistema	24
2.2 Modelo de dominio	24
2.2.1 Objeto de automatización	24
2.2.2 Glosario de términos del dominio	25
2.2.3 Diagrama de clases del modelo del dominio	25
2.3 Requisitos funcionales	25

2.4	Requisitos No Funcionales.....	27
2.5	Definición de los casos de uso	28
2.5.1	Definición de los actores	28
2.6	Diagrama de casos de usos del sistema	29
2.6.1	Descripción de los casos de uso del sistema	29
Capítulo 3: Diseño del sistema		50
3.1	Flujo de Trabajo de análisis y diseño.	50
3.2	Modelo de análisis.....	50
3.2.1	Diagrama de clases del análisis	50
3.3	Modelo de diseño.....	55
3.3.1	Clases del diseño	55
3.3.2	Patrones de diseño	55
3.3.3	Diagrama de clases del diseño	56
3.4	Diseño de base de datos	60
3.5.1	Modelo de datos	60
3.5.2	Descripción de las tablas	61
Capítulo 4: Implementación.....		66
4.1	Implementación.....	66
4.1.1	Modelo de despliegue	66
4.2	Diagrama de componentes	67
Conclusiones		68
Referencias Bibliográficas		70

Introducción

La Educación es mucho más que la transmisión de conocimientos, un recetario de procedimientos o un inventario de valores. La educación es el valioso medio para transformar su realidad y crear una sociedad más justa, con la colaboración de personas capaces y competentes.[1]

En las universidades e instituciones educativas cubanas, los proyectos educativos surgen con el objetivo de cultivar los valores en los futuros profesionales, prepararlos para la vida y que estén comprometidos con los procesos sociales, políticos y culturales del país. Formando así graduados con una alta profesionalidad y de una eticidad marcada, capaz de demostrarlo lo aprendido en el entorno comunitario donde preste sus servicios.

La Universidad de las Ciencias Informáticas (UCI) como institución académica de excelencia tiene como misión no solo formar ingenieros preparados con rigor científico, amantes del conocimiento y de la ciencia. Sino también mediante los proyectos educativos poder cultivar en sus egresados los valores que determinan la calidad de un revolucionario de estos tiempos que caractericen el profesional cubano.

Estos últimos, son un fuerte proceso que se lleva a cabo cada año, razón por la cual el colectivo de año realiza un diagnóstico a los estudiantes, teniendo en cuenta los objetivos a complementar en ese año y las prioridades de la facultad, la universidad y del territorio para la formación del profesional. La labor realizada dentro de la facultad 7, relacionada con la formación de valores y la ética profesional parte desde los proyectos educativos que se elaboran en cada brigada estudiantil. Lo que tiene como elemento central el desempeño ejercido por el colectivo de año, como la célula fundamental de la carrera así como el profesor guía y el coordinador.

En los proyectos educativos de cada brigada de la facultad se gestiona varios tipos de información entre las que se encuentran los objetivos de cada año, los problemas que afectan la integridad de la brigada y las acciones a realizar para contrarrestar los mismos.

La necesidad de encauzar integralmente la labor educativa y político-ideológica, ha conllevado a que la dirección y organización de la UCI cuente con tres dimensiones fundamentales que se desarrollan en la vida universitaria, las cuales se convierten en campos de acción donde se ejerce un sistema de influencias dirigidas a la formación, desarrollo de valores, conocimientos y habilidades.[2]

Cada una de estas dimensiones que conforman e integran el proyecto educativo son:

- ❖ Dimensión curricular
- ❖ Dimensión de extensión universitaria
- ❖ Dimensión socio-político

Por la gran importancia que poseen los proyectos educativos para la formación tanto ética como profesional del estudiante egresado de la UCI, por la necesidad de formar al profesional competente con una visión humanista y comprometida que lo conduzca a ser parte activa del proceso revolucionario, sustentadas en una concepción científica y dialéctico-materialista, es que se hace necesario la realización de un sistema para perfeccionar las actividades que se realizan en estos proyectos, que forman parte del objetivo fundamental de esta institución.

Diseñar un proyecto educativo significa planear un proceso para alcanzar una meta educativa, objetivos de aprendizaje. Esto implica desde la selección del problema surgido en un contexto educativo particular, su tratamiento hasta la presentación del informe. En otros términos, corresponde la realización de varias etapas interrelacionadas de concepción, planeamiento, formulación de acciones, implementación y evaluación. Este cronograma de trabajo, consiste en estructurar el trabajo o una secuencia de actividades que permita el desarrollo y logro de la meta propuesta de una brigada determinada.

Actualmente, en la Facultad 7, los proyectos educativos de las brigadas son elaborados de forma semi-informatizada, en documentos de tipo word o excel, lo cual conlleva en la mayoría de los casos, a las siguientes dificultades:

- ❖ Posible pérdida de la información de los proyectos educativos que se hayan realizado en las brigadas debido a algún fallo técnico en el servidor donde se guarde esta información.
- ❖ Se vuelve un trabajo engorroso para el profesor guía y demás factores que elaboran el proyecto a la hora de estructurar el contenido, ejemplo, la creación de tablas entre otros aspectos a tener en cuenta en un documento Word.
- ❖ Al coordinador del año se le dificulta conocer las actividades de cada brigada en un período de tiempo determinado, para hacerlo debe consultar documento por documento de cada una.(trabajo engorroso)

- ❖ No existe un control exacto de las acciones del mismo en las brigadas.
- ❖ Las actividades planificadas no están todo el tiempo visibles a los estudiantes y profesores de la brigada.
- ❖ No existe un estándar para este proceso en la Facultad.
- ❖ No se le da seguimiento a los problemas pendientes del curso que termina en el curso venidero.

Dada la problemática anteriormente planteada se tiene como **problema a resolver**

¿Cómo facilitar la gestión de la información que generan los proyectos educativos, realizados en las brigadas docentes de la Facultad 7?

La investigación tiene como **objeto de estudio** los sistemas de gestión académica y como **campo de acción** la gestión de los proyectos educativos en la Facultad 7 de la UCI.

Para la solución del problema se plantea como **objetivo general** desarrollar una aplicación que facilite los procesos que se realizan dentro de los proyectos educativos, de las brigadas docentes de la Facultad 7.

Para darle cumplimiento al objetivo general se tienen las siguientes **tareas de la investigación**:

- ❖ Realizar un análisis sobre la forma en que se gestiona y controla la información de los proyectos educativos.
- ❖ Analizar los sistemas informáticos existentes que gestionan la información de los proyectos educativos.
- ❖ Identificar las tecnologías que se tendrán en cuenta para el desarrollo de la aplicación.
- ❖ Elaborar la documentación correspondiente a los flujos de trabajo propuestos por la metodología seleccionada.
- ❖ Implementar las funcionalidades correspondientes al sistema que se propone, para darle solución a las necesidades actuales del cliente.

A continuación se presenta la estructura capitular del documento.

Capítulo 1: Fundamentación teórica: Se brinda una panorámica sobre la situación problemática del tema a tratar y se le hace un análisis al mismo para proporcionar la solución que se necesita, a la facultad, así como la descripción del ambiente de desarrollo empleado para la realización del nuevo sistema informático.

Capítulo 2: Características del sistema: Se describe el funcionamiento del negocio a través del diagrama, modelo de dominio y la descripción textual de los términos empleados en el mismo. Se definen además los requerimientos funcionales y no funcionales del sistema, a través del diagrama de casos de uso del sistema y las descripciones de los casos de uso para comprender mejor el funcionamiento de la aplicación.

Capítulo 3: Diseño del sistema: Está relacionado con el diseño del sistema, se identifican las clases del diseño con el propósito de refinar los requisitos funcionales identificados con anterioridad. Se construye además el modelo del diseño para aplicaciones web, lo cual constituye una ayuda para la fase de elaboración.

Capítulo 4: Implementación: Está enfocado al flujo de trabajo implementación para dar solución a los requerimientos funcionales identificados. Se construye el diagrama de componentes.

Capítulo 1 Fundamentación Teórica

En este capítulo se abordan los objetivos que persiguen los proyectos educativos para las brigadas y la idea central por la que surgen. Se hace referencia al estado del arte de los sistemas automatizados existentes para llevar a cabo el control y el seguimiento de las actividades planificadas en los mismos, tanto en el ámbito nacional como internacional. Se realiza un estudio descriptivo de las tendencias, tecnologías, metodologías y software más usados en la actualidad y se concluye con la selección de las herramientas a utilizar durante el desarrollo de la aplicación.

1.1 Concepto básico asociado al negocio

Proyecto: El término proyecto se deriva de los términos latinos PROICERE y PROIECTARE que significan arrojar algo hacia delante. Por lo que su significado en sentido genérico, es que se basa en la planeación y organización de todas las tareas y actividades. [3]

Proyecto educativo: Dado el concepto anterior se puede definir que un proyecto educativo va a estar centrado en alcanzar una meta educativa y un objetivo de aprendizaje, además que resolverá de forma organizada y planificada, un problema previamente identificado en su realidad educativa, lo que incluiría desde la selección del problema surgido en un contexto educativo particular y su tratamiento, hasta la presentación del informe. [4]

1.2 Estado del arte

1.2.1 Sistema automatizado Enter The Group

Se realizó una investigación acerca de los sistemas automatizados que puedan existir tanto a nivel nacional como internacional, que se encarguen de gestionar los proyectos educativos, se encontró el Enter the Group.

Aplicación informática creada para aprendices y profesores que quieran trabajar en proyectos en equipo, con ella, se pueden crear grupos de trabajo colaborativos y aulas virtuales, administrar reuniones y eventos a través de un calendario en línea, compartir archivos, crear encuestas, acceder a un foro de preguntas y respuesta, compartir blogs , administrar tareas, eventos, proyectos de aula, una infinidad de acciones orientadas a aumentar la productividad de nuestro trabajo a la hora de organizar nuestra vida académica. Este sistema está compuesto por una plataforma, con la cual se pueden crear estos grupos de trabajo en diferentes salas virtuales administrando las reuniones y eventos programados en el calendario de la plataforma con las capacidades y funciones necesarias para compartir archivos, crear las encuestas

y realizar una extensa cantidad de tareas propias de entornos corporativos en los cuales se están tratando temas de aprendizaje. [5]

Es una de las plataformas en términos generales más completas y recomendables para necesidades de estas características complementándose con una disponibilidad totalmente libre al momento de ser utilizada. Pero sin embargo tiene como característica principal que por ser un sistema online, el usuario necesita de una cuenta de Internet para trabajar en el, por el contrario para obtenerlo se necesita de un licencia, la cual lo convierte en un sistema privado. [6]

1.2.2 Propuesta de software

Dada la investigación que se le ha realizado al sistema automatizado existente para gestionar los proyectos educativos, se ha llegado a la conclusión que aún presentando este sistema características favorables, por las cuales pudiera utilizarse en la facultad para gestionar los proyectos educativos. Presenta también características que impiden utilizarlo, entre las que se encuentran, que es un sistema privado que se rige por una licencia y que todos los usuarios que trabajen con él necesitarían de una cuenta de Internet, por lo que los estudiantes de algunos años, como primero y segundo no lo podrían utilizar, por las racionalización de los recursos de Internet que se aplica como política en la Universidad, aunque este sistema presente una libre disponibilidad en el momento de ser usado.

Otra característica y la más desfavorable es que debido a que es un sistema online, toda la información que ahí se registre, estará expuesta en la red mundial, por lo que no tendrá seguridad ninguna y se estaría violando una de las políticas de seguridad más importante implantada en la UCI, que prohíbe tener ningún tipo de información de la universidad registrada en Internet.

Por otra parte, la ausencia de una herramienta amigable que permita el buen desarrollo de los proyectos educativos que toman lugar en la facultad, complica a su vez la labor de los trabajadores que llevan a cabo este importante proceso. Así como a los estudiantes que son evaluados y ayudados en el mismo, es por ello que se hace inminente la creación de un sistema propio de la facultad que reúna todos los requisitos para lograr el control y seguimiento de los proyectos educativos realizados en esta.

Ante este panorama, se planteó la necesidad de crear por la misma facultad una herramienta que se pueda utilizar no solo en esta sino en toda la universidad, que ayude a gestionar los procesos de un proyecto educativo desde gestionar las actividades de cada brigada y definir los problemas que la afectan en su integración como tal, hasta resolverlos, además de gestionar a cada uno de sus estudiantes tanto docentemente como moralmente para así ayudarlo en su integración como un universitario cubano.

1.3 Tecnologías y Herramientas

El principal objetivo de este trabajo de diploma es realizar un sistema Web dinámico, que sea amigable entendible y que le facilite el trabajo a sus usuarios. Para lograr esta meta se estudiaron algunas herramientas que pueden servir y ayudar en la confección del mismo, así como definir cuál de ellas sería más factible y flexible para agilizar y organizar el desarrollo del sistema educativo, que cumpla con cada unas de las soluciones previstas para cada unas de las necesidades que le dieron vida y definieron este trabajo de diploma.

1.3.1 Metodologías de desarrollo de software

Para garantizar la calidad de un software durante el desarrollo de este, es vital seguir un conjunto de pasos, que llevan a utilizar una metodología de desarrollo de software. Actualmente el desarrollo de software ha alcanzado un alto nivel debido a la competencia que existe mundialmente y más cuando a cada segundo se va perfeccionando más las tecnologías y con ellas nuevos métodos que garanticen el producto final, por lo que los desarrolladores se han visto en la necesidad de buscar técnicas mediante las cuales se logren estandarizar el trabajo de las aplicaciones que se desarrollan.

Teniendo en cuenta lo anterior se ha creado un conjunto de metodologías para el desarrollo del software que hace posible que todo el personal de un proyecto se vincule y pueda entenderse. Las metodologías se pueden agrupar en tradicionales o formales, y ágiles. Las tradicionales centran su atención principalmente en desarrollar una documentación completa de los proyectos así como el cumplimiento de un plan de proyecto, el cual debe ser definido al comienzo del mismo. Las de tipos ágiles proponen como de vital importancia la respuesta a los cambios y en mantener una buena relación con el cliente.

1.3.1.1 Programación extrema (XP)

La metodología Extreme Programming (XP) o Programación extrema, es la metodología ágil más destacada y con más aceptación en la comunidad internacional de desarrollo. Los fundamentos de la misma según su creador son: mejorar la comunicación, buscar la simplicidad, buscar retroalimentación, en que también va el trabajo y que siempre hay que proceder con valentía. Considera un aspecto natural, inevitable e incluso deseable, los cambios de requerimientos sobre la marcha. [7]

Una de las ventajas que presenta esta metodología y uno de sus requerimiento como tal es tener al cliente disponible durante todo el proyecto , determinando qué construir y cuándo, estableciendo las pruebas y las funcionalidades, o sea no solamente apoyará a los desarrolladores, sino que formará parte del grupo de trabajo. Al estar el cliente en todo el proceso, puede prevenir a tiempo de situaciones no

deseables, o de funcionamientos que no eran los que en realidad se deseaban. En otras metodologías, estas situaciones son detectadas en forma muy tardía del ciclo de desarrollo, y su corrección puede llegar a ser muy complicada. El involucramiento del cliente es fundamental para que pueda desarrollarse un proyecto con la metodología XP.

Entre las herramientas más importantes de esta metodología se encuentra, el desarrollo orientado a pruebas, que utiliza las pruebas unitarias como eje de todo desarrollo. Las interacciones suelen ser muy cortas y se promueve a los programadores a buscar soluciones y experiencia con ellas, programar sin miedo a descomponer el sistema. Se considera como una adopción de las mejores metodologías de desarrollo teniendo en cuenta lo que se pretende realizar con el proyecto, y de manera dinámica aplicarlo durante el ciclo de vida del software. [8]

XP presenta algunas desventajas, como es la dificultad de estimar cuánto va a costar un proyecto, debido a que el alcance del mismo no está completamente definido al comienzo, y es expresamente abierta a los cambios durante todo el proceso, razón por la cual se torna sumamente difícil poder realizar un presupuesto previo. Este punto puede no ser crítico para desarrollos “in house” (“en Casa”), pero para empresas desarrolladoras de software, dónde deben presupuestar si sería un gran riesgo. [9]

1.3.1.2 SCRUM

Scrum es una metodología ágil de desarrollo de proyectos que toma su nombre y principios de los estudios realizados sobre nuevas prácticas de producción por Hirotaka Takeuchi e Ikujiro Nonaka a mediados de los 80. Aunque surgió como modelo para el desarrollo de productos tecnológicos, también se emplea en entornos que trabajan con requisitos inestables y que requieren rapidez y flexibilidad; situaciones frecuentes en el desarrollo de determinados sistemas de software. Ésta es una metodología de desarrollo muy simple, que requiere trabajo duro porque no se basa en el seguimiento de un plan, sino en la adaptación continua a las circunstancias de la evolución del proyecto. [10]

Es un proceso ágil y liviano que sirve para administrar y controlar el desarrollo de software. El desarrollo se realiza en forma iterativa e incremental (una iteración es un ciclo corto de construcción repetitivo). Cada ciclo o iteración termina con una pieza ejecutable que incorpora una nueva funcionalidad. En esta metodología el equipo se enfoca solamente en construir software de calidad. La gestión de proyecto se focaliza en definir las características del software a construir y remover cualquier obstáculo que pudiera entorpecer el trabajo del equipo de desarrollo. Se busca que los equipos de desarrollo sean lo más productivos posible. [11]

Es un proceso donde se aplican de manera regular un conjunto de mejores prácticas para trabajar en equipo y obtener el mejor resultado posible de un proyecto. Estas prácticas se apoyan unas a otras y su selección tiene origen en un estudio de la manera de trabajar de equipos altamente productivos.

Se realizan entregas parciales y regulares del resultado final del proyecto, priorizadas por el beneficio que aportan al receptor del proyecto. Por ello, Scrum está especialmente indicado para proyectos en entornos complejos, donde se necesita obtener resultados pronto, donde los requerimientos son cambiantes o poco definidos, donde la innovación, la competitividad y la productividad son fundamentales.

También se utiliza para resolver situaciones en que no se está entregando al cliente lo que necesita, cuando las entregas se alargan demasiado, los costes se disparan o la calidad no es aceptable, cuando se necesita capacidad de reacción ante la competencia, cuando la moral de los equipos es baja y la rotación alta, cuando es necesario identificar y solucionar ineficiencias sistemáticamente o cuando se quiere trabajar utilizando un proceso especializado en el desarrollo de producto.

1.3.1.3 Proceso Unificado de Rational (RUP)

Es una metodología formal o proceso de ingeniería de software que provee un enfoque para asignar tareas y responsabilidades dentro de una organización de desarrollo. Su objetivo es asegurar la producción de software de alta calidad que satisfaga la necesidad del usuario final dentro de un tiempo y presupuesto previsible, además de ser uno de los procesos más generales que existe, esta enfocado a cualquier tipo de proyecto así no sea de software, se basa en la documentación generada en cada uno de sus cuatro fases:

1. Inicio (puesta en marcha)
2. Elaboración (definición, análisis y diseño)
3. Construcción (implementación)
4. Transición (fin del proyecto y puesta en producción)

RUP mejora la productividad del equipo, pues permite que cada miembro del grupo sin importar su responsabilidad específica acceda a la misma base de datos de conocimiento. Esto hace que todos compartan el mismo lenguaje, la misma visión y el mismo proceso acerca de cómo desarrollar software. Describe detalladamente todas las actividades, roles, responsabilidades, productos de trabajo y herramientas para definir el quién, qué, cuándo y cómo se realizan los procesos durante el desarrollo de un sistema de software. Representa un ideal de referencia para todo el equipo de trabajo, además es un proceso de desarrollo de software que recoge un conjunto de actividades, las cuales son necesarias para transformar los requerimientos del cliente en un sistema de software. Aparte de estar dividido en fases,

está basado en componentes y posee tres características esenciales, dirigido por casos de uso, centrado en la arquitectura y es iterativo e incremental. [12]

Divide el proceso en 4 fases, mencionadas anteriormente, de las cuales se realizan varias iteraciones según el proyecto y en las que se hace un mayor énfasis en las distintas actividades. Ésta metodología aplica 6 prácticas principales para el desarrollo de sistemas. Estas prácticas son:

- ❖ Desarrollo de software en forma iterativa: permite ir creciendo en el entendimiento del problema a través de refinamientos sucesivos. Esto también permite introducir cambios tácticos en los requerimientos, características del sistema o en los tiempos.
- ❖ Gestión de requerimientos: es necesario para garantizar que al final el software tenga la calidad requerida.
- ❖ Uso de arquitecturas basadas en componentes: esta hace que el sistema pueda reutilizar componentes desarrollados con antelación y facilitar el trabajo que se realiza.
- ❖ Modelización visual del software: el proceso le demuestra cómo modelar visualmente software para capturar la estructura y el comportamiento de arquitecturas y de componentes.
- ❖ Verificación de calidad del software: RUP le asiste en el planeamiento, el diseño, la puesta en marcha, la ejecución, y la evaluación de las pruebas de confiabilidad, funcionalidad, performance de la aplicación y el sistema.
- ❖ Control de cambios: la capacidad de manejar los cambios asegurándose que cada cambio sea aceptable, y pudiendo continuar con los mismos es esencial en un ambiente en el cual el cambio es inevitable.

1.3.2 Lenguaje de modelado

Se pueden capturar las partes esenciales de un software mediante el uso de las notaciones gráficas en la modelación de una aplicación de software. El lenguaje de modelación es independiente del lenguaje de programación que se desea emplear en el desarrollo del software. Con el transcurso de los años se ha producido un auge en la Programación Orientada a Objeto (POO), lo que ha conllevado a la creación de diferentes lenguajes de modelado orientado a objetos. [13]

1.3.2.1 Lenguaje de modelado OO i*

La notación i* fue creada en la primera mitad de la década de los 90. Permite expresar de forma clara y sencilla los objetivos de los actores que aparecen en los modelos y la dependencia entre ellos.

Consta con una notación gráfica que permite tener una visión intuitiva y unificada del entorno modelado mostrando tales actores y dependencias. Tiene la desventaja de no tener una definición única del lenguaje. Además, las definiciones existentes no son tan claras como se desearía ya que contienen ambigüedades y contradicciones.

1.3.2.2 Lenguaje de modelado OO UML

UML es un lenguaje que provee un conjunto de herramientas que permiten modelar (analizar y diseñar) sistemas orientados a objeto. Este lenguaje tiene una notación gráfica muy expresiva que permite representar en mayor o menor medida todas las fases de un proyecto informático: desde el análisis con los casos de uso, el diseño con los diagramas de clases, objetos, etc., hasta la implementación y configuración con los diagramas de despliegue. Es uno de los lenguajes de modelado de sistemas de software más conocido y utilizado en la actualidad; está respaldado por el OMG (Object Management Group). [14]

Las herramientas que provee UML para el desarrollo de sistemas OO son:

- ❖ Diagrama de casos de uso.
- ❖ Diagrama de clases.
- ❖ Diagrama de estado.
- ❖ Diagrama de secuencia.
- ❖ Diagrama de actividades.
- ❖ Diagrama de colaboración.
- ❖ Diagrama de componentes.
- ❖ Diagrama de distribución.

Características de UML:

- ❖ Permite adaptarse fácilmente a los usuarios, así como a otros usuarios de otros métodos.
- ❖ Permite modelar nuevos elementos en el proyecto que se desarrolla.
- ❖ Provee una expresividad e integridad holística mejorada, respecto a otros lenguajes de modelamiento visual.
- ❖ Es fácil de aprender y usar, ya sea respecto a las técnicas más avanzadas, es decir, estereotipos y propiedades, así como algunos cambios en la anotación y semánticas.
- ❖ Será la elección obvia para realizar nuevos proyectos, especialmente cuando se incremente la disponibilidad de herramientas, libros y cursos.

1.3.3 Framework de desarrollo

Los framework de desarrollo han sido un gran paso de avance para la creación de sistemas de software. En términos generales se definen como un conjunto estandarizado de conceptos, prácticas y criterios para enfocar un tipo de problemática particular, lo que llevado a término informático está comprendido como la estructura conceptual y tecnológica de soporte, definida normalmente con artefactos o módulos de software concretos, con base en la cual otro proyecto de software puede ser organizado y desarrollado. Dentro de él se pueden encontrar soportes de programas, bibliotecas y un lenguaje interpretado entre otros software para facilitar el desarrollo y unir las diferentes estructuras y componentes de un proyecto.

Esta importante estructura de soporte definida, en la que otro proyecto de software puede ser organizado y desarrollado, ha sido diseñada con la intención de facilitar el desarrollo de software, permitiendo a los diseñadores y programadores pasar más tiempo identificando requerimientos de software que tratando con los tediosos detalles de bajo nivel de proveer un sistema funcional, se han convertido en la piedra angular de la moderna ingeniería del software, con la cual se puede añadir las últimas piezas para construir un sistema concreto.

1.3.3.1 Symfony

Es un framework para aplicaciones Web desarrolladas en PHP5 por Fabien Potencier y patrocinadas por los laboratorios de Sensio, agencia francesa Web muy conocida por sus puntos de vista innovadores en el desarrollo Web. Acelera la creación y mantenimiento de las aplicaciones Web, y permite la reutilización de código, lo cual agiliza el desarrollo de la aplicación. Su diseño permite optimizar las aplicaciones y mantener una organización de la misma. Permite reducir el tiempo de desarrollo de una aplicación, pues proporciona varias herramientas y clases las cuales puedes ser reutilizada por los desarrolladores. Permite el desarrollo por capas, ya que se basa en el patrón arquitectónico modelo vista controlador (MVC).

1.3.3.2 Kumbia

Es un framework para desarrollo de aplicaciones Web escrito en PHP5. Fomenta la eficiencia y velocidad en la creación y mantenimiento de aplicaciones Web, reemplazando códigos repetitivos. Intenta proporcionar un medio fácil para el desarrollo de aplicaciones robustas para entornos educativos y comerciales, gracias a que es flexible y configurable.

1.3.3.3 CakePHP

Es un framework para aplicaciones Web desarrollado en PHP, creado sobre los conceptos de “Ruby on Rails”. Empezó su desarrollo en el 2005 cuando Ruby On Rails estaba ganando popularidad y desde entonces ha generando muchos proyectos y su comunidad ha crecido. Facilita al usuario la interacción con las bases de datos mediante el uso de Active Record, además se basa en el patrón arquitectónico modelo vista controlador (MVC), lo que le permite dividir la aplicación en capas. Tiene como características fundamentales:

- ❖ Compatible con PHP4 y PHP5.
- ❖ CRUD de la base de datos integrado.
- ❖ URLs amigables.
- ❖ Sistema de plantillas rápido y flexible.
- ❖ Helpers para AJAX, Java script, HTML, forms y más.
- ❖ Trabaja en cualquier subdirectorio del sitio.
- ❖ Validación integrada.
- ❖ Scaffolding de las aplicaciones.
- ❖ Access control lists.
- ❖ Sanitización de datos.
- ❖ Componentes de seguridad y sesión.

1.3.3.4 Comparación entre los framework estudiados[15]

Características	Kumbia	Symfony	Cake
MVC Nativo	X	X	X
Enrutamiento avanzado	X	X	X
Scaffold (Generadores de código)	X	X	X
Scaffold avanzado (Generadores de formularios)	X		
Correo electrónico	X	X	X
Mapeo objeto-relacional	X	X	X

Asociaciones ORM	X	X	X
Eventos ORM	X	X	X
Sistema de plantillas	X	X	X
Integración con Smarty	X	X	X
Generación de reportes (múltiples formatos)	X		
HTML Helpers	X	X	X
Plug-Ins (Integración terceros)	X	X	X
i18n (Internacionalización)		X	X
Efectos visuales	X	X	X
Integración AJAX	X	X	X
Componente de sesiones	X	X	X
Loggers ¹	X	X	
Documentación en inglés		X	X
Documentación en español	X	X	
ACL (Access Control Lists) ²	X	X	X
Soporte MySQL	X	X	X
Soporte postgreSQL	X	X	X
Soporte oracle	X	X	

¹ Encargado de loguear lo necesario en la aplicación para dar información al usuario acerca del estado interno del framework.

² Listas de control de acceso para controlar el acceso a ciertos objetos protegidos mediante la administración de privilegios.

Soporte SQL server	X		
Soporte SQLite	X	X	

1.3.4 Arquitectura

En los inicios de la informática, la programación se consideraba un arte y se desarrollaba como tal, debido a la dificultad que entrañaba para la mayoría de las personas, pero con el tiempo se han ido descubriendo y desarrollando formas y guías generales, con base a las cuales se puedan resolver los problemas. A estas, se les ha denominado arquitectura de software que se define como el diseño de más alto nivel de la estructura de un sistema que está conformado por un conjunto de patrones y abstracciones coherentes que proporcionan el marco de referencia necesario para guiar la construcción del software para un sistema de información.

1.3.4.1 Patrón modelo vista controlador (MVC)

Este patrón de modelo que fue descrito por primera vez en 1979 por Trygve Reenskaug de Smalltalk, en unos laboratorios de gran investigación de Xerox, hoy uno de los más utilizados mundialmente, describe una forma muy utilizada en la web de organizar el código de una aplicación, formado por tres componentes modelo, vista, y controlador. El modelo representa la información con la que trabaja la aplicación, es decir, la lógica del negocio. La vista transforma el modelo en una página web que permite al usuario interactuar con ella. El controlador se encarga de procesar las interacciones del usuario y realiza los cambios apropiados en el modelo o en la vista. [16]

El empleo del patrón proporciona varias ventajas para el desarrollo de la aplicación, por ejemplo:

- ❖ Las vistas proveen mayor flexibilidad y agilidad.
- ❖ Se puede crear múltiples vistas de un modelo.
- ❖ Las vistas pueden anidarse.
- ❖ Se puede cambiar el modo en que una vista responde al usuario sin cambiar su representación visual.
- ❖ Se puede sincronizar las vistas.
- ❖ Facilita el mantenimiento del sistema.

1.3.5 Lenguaje de programación para la Web

En la actualidad existen diferentes lenguajes de programación, estos han surgido debido a las tendencias y necesidades de las plataformas existentes, con ellos se han podido desarrollar varios sistemas Web desde su surgimiento, lo cual ha servido de ayuda en el desarrollo mundial de las tecnologías Web, brindándole a cada uno de los usuarios una web dinámica, fácil de usar y permite el intercambio entre usuario y máquina. Estos lenguajes surgen en el transcurso del tiempo con el avance de las tecnologías y el surgimiento de nuevos problemas a dar solución. A continuación se presenta una introducción a los diferentes lenguajes de programación para la web.

1.3.5.1 Lenguaje ASP.NET

ASP.NET es un conjunto de tecnologías de desarrollo de aplicaciones Web comercializado por Microsoft. Es usado por programadores para construir sitios Web domésticos, aplicaciones Web y servicios XML. Forma parte de la plataforma .NET y es la tecnología sucesora de Active Server Pages (ASP).

ASP.NET es una nueva estructura de programación y fue diseñado con el objetivo de que dichas aplicaciones respondan rápidamente a las solicitudes de los usuarios, sin importar la cantidad de datos que se estén procesando en el servidor debido a que en una página ASP se pueden incluir: HTML plano, código de scripting y texto, no hay una distinción formal entre el contenido de una página y su comportamiento. Impone un cierto orden sobre el modelo de programación estándar ASP, proporciona diversas mejoras en las cuales se destacan: Rendimiento: se compila desde el código nativo, lo que permite mucho mejor rendimiento y un almacenamiento de la caché en el servidor. Rapidez en programación: mediante diversos controles, se logra con unas pocas líneas y en menos de 5 minutos mostrar toda una base de datos y hacer rutinas complejas. Servicios Web: trae herramientas para compartir datos e información entre distintos sitios. Seguridad: tiene diversas herramientas que garantizan la seguridad de nuestras aplicaciones.[17]

1.3.5.2 Lenguaje PHP

Procesador de hipertexto, es gratuito e independiente de plataforma, rápido, con una amplia librería de funciones y mucha documentación. Fue creado en 1994 por Rasmus Lerdorf. Se ejecuta del lado servidor Web, pero puede ser utilizado desde una interfaz de líneas de comandos o en la creación de otros programas como aplicaciones de interfaz gráfica, mediante el uso de bibliotecas Qt o GTK+. Fue diseñado originalmente para la creación de páginas Web.

Es un lenguaje orientado a objeto, lo que permite la reutilización de código y un rápido desarrollo. Posee compatibilidad con las bases de datos más comunes, como MySQL, Oracle y PostgreSQL entre otros, tiene como característica que es fácil de aprender, y está pensado para desarrollar páginas web eficientes. Se utiliza para escribir el contenido dinámico de las páginas. Soportado por la mayoría de los servidores web entre ellos Apache. Se utiliza en cualquier sistema operativo. Permite manipular la información almacenadas en la base de datos como MySQL. Proporciona soporte para diferentes protocolos de comunicación conocidos entre los cuales se tienen:

- ❖ HTTP (Protocolo de transferencia de hipertexto).
- ❖ IMAP (Protocolo de acceso a mensajes de Internet).
- ❖ FTP (Protocolo de transferencia de archivos).
- ❖ LDAP (Protocolo ligero de acceso a directorios), entre otros.

Es publicado bajo la PHP License, la Free Software Foundation considera esta licencia como software libre. [18]

1.3.5.3 Lenguaje Java Script

Es un lenguaje de script no interpretado, no requiere de un compilador que se ejecuta de lado del cliente, por lo que es interpretado por el navegador y fue desarrollado por Netscape Communication. Es orientado a objeto, y permite acceder a objetos en aplicaciones Web. JavaScript ha tenido influencia de diferentes lenguajes y fue diseñado con sintaxis similar a Java, aunque más fácil de utilizar por programadores principiantes. Permite realizar las validaciones del lado del cliente, permitiendo el acceso innecesario al servidor Web, posibilitando que dicho servidor tenga más funcionalidad a la hora de responder peticiones de los usuarios. Es interpretado por la mayoría de los navegadores más comunes usados como Netscape, Internet Explorer, Mozilla Firefox, Opera, entre otros. [19]

1.3.6 Herramientas case

Las herramientas de desarrollo llegan hacer de vital importancia para el desarrollo de un sistema, que brinda facilidad de trabajo y permite modelar, organizar y crear cada una de sus funcionalidades. Para desarrollar un sistema normalmente solo es necesario un editor de texto, un intérprete o compilador y una terminal de líneas de comando, pero siempre es más rápido y fácil si se usan algunas herramientas o un Entorno de Desarrollo Integrado (IDE) simplificando el trabajo y ahorrando tiempo de desarrollo. Al seleccionar una herramienta, es importante que cumpla con los requerimientos que se necesitarían para desarrollar el software, en esto se debe tener en cuenta la tecnología que se va a usar y la plataforma de desarrollo.

1.3.6.1 Rational rose

El Rational rose es una herramienta de modelado visual con plataforma independiente, que permite la comunicación entre los miembros del equipo. Tiene como ventaja que utiliza la notación estándar en la arquitectura de software (UML), posibilita a los arquitectos y desarrolladores de software visualizar todo el sistema mediante el uso de un lenguaje común. Mantiene la consistencia de los modelos del sistema de software. Permite chequear las sintaxis UML y realizar ingeniería inversa. Posibilita la generación de documentación automáticamente y la generación de código a partir de los modelos. A través de él los diseñadores pueden modelar sus componentes e interfaces en forma individual y luego unirlos con los demás componentes del proyecto. [20]

1.3.6.2 Visual paradigm

Visual paradigm para UML es una herramienta UML profesional que soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientados a objetos, construcción, pruebas y despliegue. El software de modelado UML ayuda a una más rápida construcción de aplicaciones de calidad, mejores y a un menor coste. Permite dibujar todos los tipos de diagramas de clases, código inverso, generar código desde diagramas y generar documentación. La herramienta UML CASE también proporciona abundantes tutoriales de UML, demostraciones interactivas de UML y proyectos UML. [21]

Posibilita la representación gráfica de los diagramas permitiendo ver el sistema desde diferentes perspectivas, como el de componentes, despliegue, secuencia, casos de uso, clase, actividad, estado, entre otros. Además, identifica requisitos y comunica información, se centra en cómo los componentes del sistema interactúan entre ellos, sin entrar en detalles excesivos, también, permite ver las relaciones entre los componentes del diseño y mejora la comunicación entre los miembros del equipo usando un lenguaje gráfico.

Tiene disponible distintas versiones: Enterprise, Professional, Standard, Modeler, Personal y Community. Facilita licencias especiales para fines académicos. Entre sus características más significativas se puede resaltar que su licencia es gratuita, posee varios idiomas, sus ediciones son compatibles y es de fácil uso para la creación de aplicaciones web. [22]

1.3.7 Entorno de desarrollo

Un entorno de desarrollo integrado (IDE) es un programa informático compuesto por un conjunto de herramientas de programación. Puede dedicarse en exclusiva a un sólo lenguaje de programación o bien, puede utilizarse para varios. Un IDE es un entorno de programación que ha sido empaquetado como un

programa de aplicación, es decir, consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica (GUI), con el que se puede crear distintos tipos de aplicación.

1.3.7.1 Net Beans

Es un IDE, libre, de código abierto (Open-Source) para desarrolladores de software. Presenta todas las herramientas que se necesita para crear aplicaciones de escritorio profesionales, aplicaciones Web y aplicaciones móviles con el lenguaje Java, C / C + +, e incluso los lenguajes dinámicos como PHP, Javascript, Groovy y Ruby. Es fácil de instalar y utilizar directamente, se ejecuta en varias plataformas, incluyendo Windows, Linux, Mac OS X y Solaris. Posee un plug-ing que permite desarrollar aplicaciones Web en PHP, y la versión actual trae incorporado soporte para el desarrollo en PHP utilizando el framework de Symfony, posee muchas funciones que hacen posible un mejor desarrollo con este framework.

Esta herramienta permite que los programadores puedan escribir, compilar, depurar y ejecutar programas. Está escrito en Java, pero puede servir para cualquier otro lenguaje de programación como anteriormente se plantea con especificidad. Existe además un número importante de módulos para extender el NetBeans IDE. Es un producto libre y gratuito sin restricciones de uso. Dispone de soporte para crear interfaces gráficas de forma visual, desarrollo de aplicaciones web, control de versiones, colaboración entre varias personas, creación de aplicaciones compatibles con teléfonos móviles, resaltado de sintaxis y por si fuera poco sus funcionalidades son ampliables mediante la instalación de packs. [23]

1.3.7.2 ZendStudio for Eclipse

ZendStudio 6.0 es un IDE de desarrollo para aplicaciones de la Web 2.0, gratuito y de código fuente abierto. Está basado en el conocido entorno de desarrollo Eclipse, también de código abierto; pero mientras que Eclipse está focalizado en el desarrollo para Java, ZendStudio es una distribución focalizada en el desarrollo Web, con soporte a HTML, CSS y JavaScript. Soporta varias librerías como: ExtJS, Yahoo UI y JQuery para la presentación así como la inclusión de framework para el desarrollo de la web pudiendo combinarlas fácilmente en una aplicación. Está disponible como una aplicación independiente, se puede encontrar para dos plataformas fundamentales: Windows y GNU/Linux. [24]

1.3.8 Sistemas gestores de base de datos

Una base de datos o banco de datos es un conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso. Existen unos programas denominados sistema gestor de bases de datos (SGBD), que permiten almacenar y posteriormente acceder a los datos de forma

rápida y estructurada. Las propiedades de estos, así como su utilización y administración, se estudian dentro del ámbito de la informática.

Los SGBD poseen grandes ventajas entre las que se destacan:

- ❖ Facilidad de manejo de grandes volúmenes de información.
- ❖ Gran velocidad de ejecución de las consultas.
- ❖ Independencia del tratamiento de información.
- ❖ Seguridad de la información (acceso a usuarios autorizados), protección de información, de modificaciones, inclusiones y consulta.

1.3.8.1 MySQL

MySQL es un servidor de bases de datos SQL de código abierto, lo que significa que cualquier programador puede remodelar el código de la aplicación para mejorarlo. Existen versiones para Linux y Windows, esto lo hace que sea multiplataforma. Este sistema licenciado bajo la GNU GPL, fue creado por la empresa sueca MySQL AB, que mantiene el copyright del código fuente del servidor SQL, así como también de la marca.

Popular herramienta de código abierto con más de 100 millones de copias de su software descargado o distribuidos a lo largo de su historia. Con su velocidad, fiabilidad y facilidad de uso, se ha convertido en la opción preferida para la Web y para empresas de telecomunicaciones, ya que elimina los problemas más importantes asociados con el tiempo de inactividad, mantenimiento, administración modernos y de aplicaciones en línea. [25]

Las principales características de este gestor de bases de datos son las siguientes:

- ❖ Aprovecha la potencia de sistemas multiprocesador a la hora de realiza las búsquedas de datos.
- ❖ Soporta gran cantidad de tipos de datos para las columnas.
- ❖ Posee API's³ que permiten el desarrollo en gran cantidad de lenguajes (C, C++, Java, PHP.).
- ❖ Gran portabilidad entre sistemas.
- ❖ Soporta hasta 32 índices por tabla.
- ❖ Gestión de usuarios y contraseñas, manteniendo un muy buen nivel de seguridad en los datos.

1.3.8.2 PostgreSQL

Es un sistema de gestión de base de datos relacional orientada a objetos de software libre, publicado bajo la licencia BSD (*Berkeley Software Distribution*).⁴ PostgreSQL es un sistema de bases de

³ interfaz de programación de aplicaciones, es el conjunto de funciones y procedimientos ó métodos, en la programación orientada a objetos, que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción. Son usados generalmente en las bibliotecas.

datos objeto-relacional con características de los mejores sistemas de bases de datos comerciales. Es libre y su código fuente completo está disponible.

Es un motor de base de datos surgido en 1986 con el lanzamiento de su primera versión. Este gestor es altamente potente y posee prestaciones y funcionalidades equivalentes a otros de carácter comercial. Es más completo que MySQL, pues métodos almacenados, restricciones de integridad y vistas. Actualmente su última versión, la 8.4 fue lanzada en julio del 2009.

Algunas de sus características principales:

- ❖ Alta concurrencia.
- ❖ Amplia variedad de tipos nativos.
- ❖ Uso de disparadores.
- ❖ Funciones de ventanas.
- ❖ Expresiones de tablas comunes y consultas recursivas.
- ❖ Instalaciones ilimitadas.
- ❖ Estabilidad y confiabilidad.
- ❖ Gran soporte a proveedores.
- ❖ Extensible.
- ❖ Multiplataforma.
- ❖ Diseño para ambientes de amplio volumen.

Realizado el estudio de las ventajas de los anteriores gestores de bases de datos se hace recomendable el uso de PostgreSQL dada sus ventajas fundamentales de uso libre y soportar varias plataformas, lo que aumenta su potencia de uso y aplicación en distintos sistemas operativos. [26]

1.4 Propuesta de herramientas para el desarrollo

Luego de realizado el estudio a las tecnologías existentes mundialmente para el desarrollo del sistema, se seleccionaron las siguientes herramientas, pues se adaptan al perfil de trabajo, y darán respuesta a las necesidades del software con una excelente calidad.

Para el desarrollo del sistema de software se propone usar la metodología RUP, la cual aplica 6 prácticas principales que documenta con mucha calidad y especificidad el proceso de desarrollo del sistema, siendo la más usada en la universidad y la más adecuada para la solución. Como lenguaje de modelado se empleará el UML en su versión 2.0, pues proporciona a los desarrolladores un mejor entendimiento y asimilación de lo que se está creando, como también agiliza la detención de errores, así

⁴ Distribución de software Berkeley es un sistema operativo derivado del sistema Unix nacido a partir de los aportes realizados a ese sistema por la Universidad de California en Berkeley.

como la resolución de los mismos y tiene como característica significativa que minimiza el tiempo invertido en el desarrollo de la arquitectura. Para visualizar el proceso que se llevara a cabo se utilizará como herramienta case Visual Paradigm UML en su versión 6.4 pues ayuda a una más rápida construcción de aplicaciones de calidad, mejores y a un menor coste.

Para su creación se propone utilizar el lenguaje de programación del lado del servidor, PHP en su versión 5.2.3 siendo este un lenguaje liviano y el más utilizado mundialmente para el desarrollo de aplicaciones Web, el cual cumple con los requisitos necesario para la creación de cada funcionalidad del sistema como por ejemplo: presenta una amplia documentación, es gratuito e independiente de la plataforma donde se quiera utilizar. Para lenguaje del lado del cliente se propone JavaScript y HTML, que permite realizar validaciones del lado del cliente, es orientado a objeto y es el más común que se utiliza para el trabajo del lado del cliente; y HTML como lenguaje etiquetado que describe la estructura y el contenido en forma de texto, el cual además brinda la posibilidad de un mejor manejo de el documento Web.

Después de analizados los frameworks existentes para PHP, se selecciona Symfony en su versión estable más reciente 1.3.3, teniendo en cuenta que posee una amplia comunidad a nivel mundial, mucha documentación en inglés y en español, una gran variedad de plug-ing, que permiten agilizar el trabajo. Es gratuito y de código abierto. Es compatible con el SGBD que se propone utilizar.

Para llevar a cabo el desarrollo en PHP se empleará el NetBeans IDE, como entorno de desarrollo integrado, para PHP en su versión 6.8, pues incorpora un módulo para el trabajo con Symfony, muy amigable y fácil de utilizar. Como sistema gestor de base de datos se utilizará MySQL en su versión 5.0.45. Es muy sencillo de utilizar, se integra perfectamente con el framework Symfony, es libre y una de las más utilizados en sistemas que no requieran de mucho procesamiento de información por parte del gestor de bases de datos. Para el diseño visual de la base de datos, modelación, creación y mantenimiento de la misma se utilizará PhpMyAdmin 2.10.2.

Para hacer más agradable el diseño de las interfaces del sistema se propone usar Ext JS⁵ en su versión 2.2 , es un framework JavaScript del lado del cliente, que permite crear aplicaciones web con interfaces muy similares a la de una aplicación de escritorio. Entre sus características se pueden encontrar que es multiplataforma, completamente orientado a objetos, y posee múltiples posibilidades para el trabajo con las validaciones y manejo de errores en el cliente.

⁵Biblioteca de JavaScript creada para el desarrollo de aplicaciones web interactivas usando tecnologías como AJAX, DHTML y DOM.

En la arquitectura del sistema se utilizará el patrón modelo vista controlador, el mismo describe una forma muy utilizada en la web de organizar el código de una aplicación.

Para el desarrollo del sistema se propone como servidor web Apache en su versión 2.2 ya que es el servidor web más utilizado hoy en día, debido a que es multiplataforma, es modular pues puede ser adaptado a diferentes entornos, necesidades y es extensible.

Conclusiones

En este capítulo se realizó una profunda investigación sobre la información que se controla en los proyectos educativos, así como las aplicaciones informáticas existentes para gestionar las actividades que se llevan a cabo en estos, tanto en Cuba como en el resto del mundo, incluso en la misma universidad UCI. Se sometieron a investigación y comparación las diferentes tecnologías, herramientas y lenguajes existentes en el mundo, que se utilizan para el desarrollo de un sistema de software referente a la Web, hasta llegar al proceso de selección, para analizar cual de ellas sería mas factible y complementaria para la creación de un sistema que controle y le de seguimiento a las actividades planificadas en los proyectos educativos.

Capítulo 2: Características del Sistema

Introducción

En este capítulo se abarca todo lo referente al modelo de dominio dentro del cual se analizarán los conceptos y entidades presentes en el entorno donde se desarrollará el sistema. Se presentará detalladamente los requerimientos funcionales y no funcionales con los que tiene que cumplir dicho sistema una vez creado, además se dará una descripción detallada de los actores involucrados en el mismo, así como los detalles de los casos de uso del sistema.

2.1 Modelo de dominio

Debido a que los proyectos educativos no son más que varios procesos que va desde la evaluación académica de un estudiante, hasta gestionar las actividades de una brigada, tanto dentro, como fuera de la Universidad. Éstos no tienen un cliente bien definido sino personas que están involucradas durante el transcurso de los mismos. Por ello, no existe una forma ó modelo único para realizarse, puesto que cada universidad, facultad o persona que lo lleva a cabo o dirige puede tener varias formas a la hora de desarrollarlo.

Por estas razones resultó difícil de encontrar procesos de negocio bien estructurados que permitieran realizar un modelo completo de dicho negocio; por consiguiente se tomó la decisión de realizar un modelo de dominio y un glosario de términos para describir los términos empleados en el modelo. Un modelo de dominio captura los tipos más importantes de objetos que existen o los eventos que suceden en el entorno donde estará el sistema, teniendo como ventaja el permitir ayudar a los usuarios, clientes y desarrolladores a utilizar un vocabulario común para poder entender el contexto en que se emplaza el sistema.

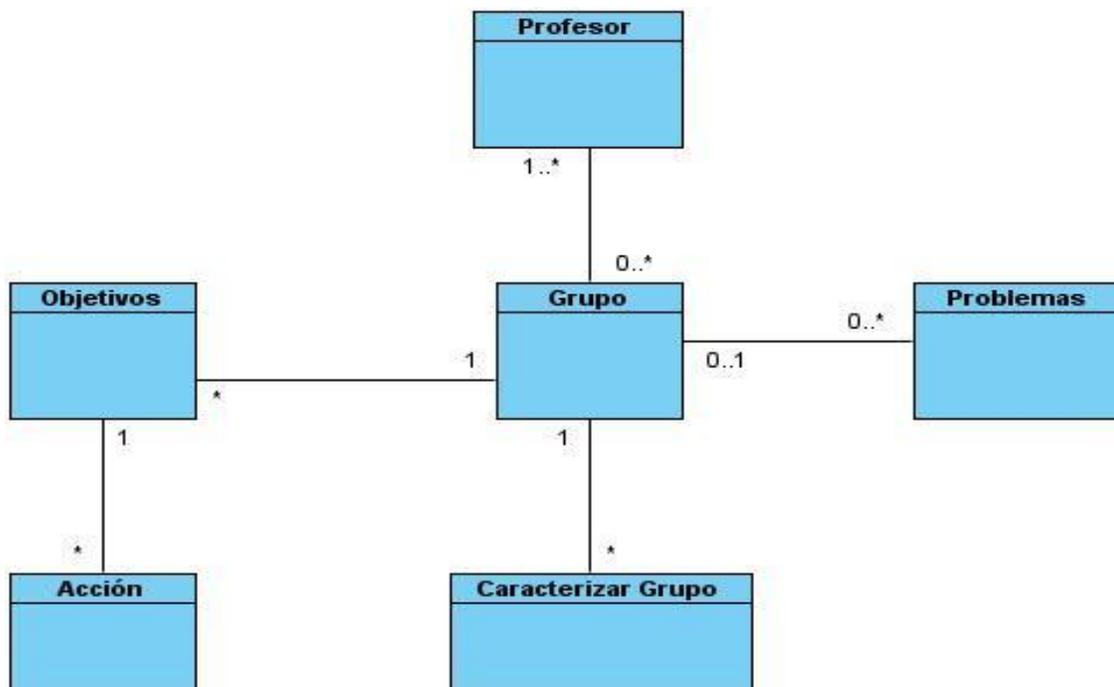
2.1.1 Objeto de automatización

Se desea automatizar el seguimiento y el control de los proyectos educativos realizados en la facultad 7, donde se pueda gestionar toda la información, para lograr un agrupamiento, organización y estructura, evitar una posible pérdida de este proceso, que pueda afectar el trabajo que se hace en cada brigada en aras de mejorar su integralidad y brindar la posibilidad a cada persona involucrada, de obtener la información guardada

2.1.2 Glosario de términos del dominio

- ❖ **Grupo:** Es el conjunto de estudiantes que comparten una ubicación académica dentro de la facultad.
- ❖ **Profesor:** Es la persona que será la encargada de desarrollar el proyecto educativo en un grupo, donde se evaluará, organizará y estructurará este proceso.
- ❖ **Objetivos:** Metas que se quieren alcanzar con el proyecto educativo, o sea acción que se debe lograr durante el ciclo de vida de este.
- ❖ **Problemas:** Situaciones detectadas en el grupo que constituyen un problema a resolver.
- ❖ **Acción:** Medidas que se toman para alcanzar los objetivos.
- ❖ **Rol:** Cargo que desempeña dentro proyecto educativo.
- ❖ **Características del grupo:** Conjunto de particularidades que presenta la brigada.

2.1.3 Diagrama de clases del modelo del dominio



2.2 Requisitos funcionales

Id	Requisito	Descripción
RF. 1	Gestionar profesor	Gestiona los profesores que van a interactuar con la aplicación.
RF. 1.1	Insertar profesor	Adiciona un nuevo profesor.
RF. 1.2	Eliminar profesor	Elimina un profesor existente.

Capítulo 2: Características del Sistema

RF. 1.3	Modificar profesor	Modifica un profesor existente.
RF. 1.4	Buscar profesor	Busca un profesor y muestra el listado de los existentes según el criterio de búsqueda.
RF. 2	Gestionar grupo	Gestiona el grupo correspondiente al proyecto educativo.
RF. 2.1	Insertar grupo	Añade un nuevo grupo.
RF. 2.2	Eliminar grupo	Elimina un grupo existente.
RF. 2.3	Modificar grupo	Modifica un grupo existente.
RF. 2.4	Buscar grupo	Busca un grupo ó muestra el listado de los grupos existentes.
RF. 3	Gestionar usuarios	Gestiona los usuarios que pueden acceder a las diferentes funcionalidades dentro de la aplicación.
RF. 3.1	Insertar usuario	Añade un nuevo usuario.
RF. 3.2	Eliminar usuario	Elimina un usuario existente.
RF. 3.3	Modificar usuario	Modifica un usuario existente.
RF. 3.4	Buscar usuario	Busca un usuario ó muestra el listado de los existentes.
RF. 4	Gestionar objetivo	Gestiona los objetivos, que estarán desarrollándose dentro del proyecto educativo.
RF. 4.1	Insertar objetivo	Añade un nuevo objetivo.
RF. 4.2	Eliminar objetivo	Elimina un objetivo existente.
RF. 4.3	Modificar objetivo	Modifica un objetivo existente.
RF. 4.4	Buscar objetivo	Busca un objetivo ó muestra el listado de los existentes.
RF.5	Gestionar acción	Gestiona las acciones, que serán las actividades propuestas para cumplir los objetivos.
RF. 5.1	Insertar acción	Añade una nueva acción.
RF. 5.2	Eliminar acción	Elimina una acción existente.
RF. 5.3	Modificar acción	Modifica una acción existente.
RF. 5.4	Buscar acción	Busca una acción ó muestra el listado de las existentes.
RF. 5.5	Reporte acción	Genera un documento donde se almacena las acciones existentes.
RF. 6	Gestionar problema	Gestiona los problemas que se detectan en la brigada.
RF. 6.1	Insertar problema	Añade un nuevo problema.
RF. 6.2	Eliminar problema	Elimina un problema existente
RF. 6.3	Modificar problema	Modifica un problema existente.

RF. 6.4	Buscar problema	Busca los problemas dado un grupo específico.
RF. 7	Registrar características del Grupo	Caracteriza los grupos registrados en la aplicación.
RF. 7.1	Insertar características	Añade una nueva característica.
RF. 7.2	Eliminar características	Elimina características.
RF. 7.3	Modificar características	Modifica características existentes.
RF. 7.4	Buscar características	Busca las características de un grupo dado.
RF. 8	Visualizar objetivos	Permite visualizar todos los objetivos dado un criterio de búsqueda.
RF. 9	Visualizar acciones	Permite visualizar todas las acciones dado un criterio de búsqueda.
RF. 10	Visualizar problemas	Permite visualizar todos los problemas dado un criterio de búsqueda.
RF. 11	Autenticar usuario	Permite autenticar en la aplicación los diferentes usuarios que interactúan en la misma.

2.3 Requisitos No Funcionales

Id	Requisito	Descripción
Usabilidad	Fácil empleo para los usuarios.	El sistema debe garantizar un acceso fácil y rápido. Está orientado a personas con conocimientos básicos de computación que poseen los permisos necesarios para interactuar con esta aplicación Web. Para una correcta explotación de las facilidades que brinda se recomienda principalmente para personas que tengan un mínimo conocimiento sobre los proyectos educativos.
Seguridad	Seguridad del sistema	La seguridad del sistema estará dada por un sistema de autenticación de usuarios, cuyos permisos están definidos según el rol que desempeñe el usuario que se autentica.
Rendimiento	Estabilidad	Las páginas estarán diseñadas con el objetivo de mantener el mismo rendimiento y estabilidad a medida que aumenta la cantidad de datos a gestionar.
Soporte	Multiplataforma	El sistema debe descansar sobre un Servidor Apache 2.2, teniendo este un soporte para PHP 5.2.3, ya que el sistema debe ser multiplataforma y este servidor nos brinda esta funcionalidad.
Restricciones de diseño	Gestor de base de datos MySQL.	El sistema debe descansar sobre el gestor de bases de datos MySQL ya que es un gestor potente, seguro, liviano y estable que pertenece al movimiento de software libre.
Restricciones de diseño	Compatibilidad con navegadores.	El sistema será compatible con navegadores como: Mozilla Firefox e Internet Explorer, en sus versiones 1.5 y 6.0 respectivamente.

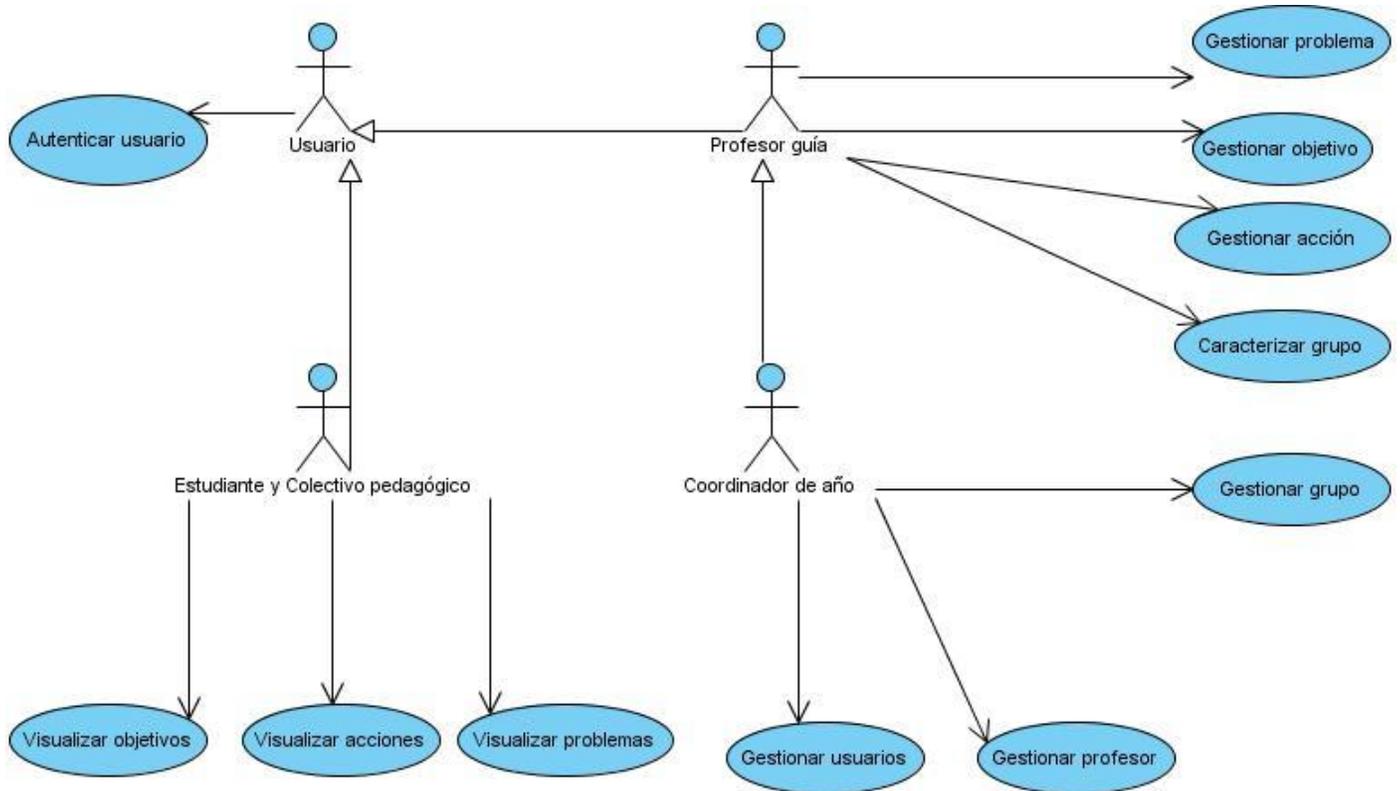
Apariencia ó Interfaz externa		La interfaz será sencilla y amigable para el usuario. Los colores a usar estarán acordes a los fines de la aplicación.
Software		<p>Para clientes: Se tendrá acceso al sistema a través del navegador Web. Firefox Mozilla 1.5 o superior.</p> <p>Para capa de presentación: El servidor debe tener PHP Versión 5.2.3 Servidor HTTP, Apache. Sistema operativo LINUX, Windows 98 o superior.</p> <p>Para capa de datos: Servidor de base de datos MySQL Versión 5.0.45.</p>
Hardware	Requerimientos mínimos para clientes	Procesador Pentium III a 1GHz de velocidad de procesamiento y 128 Mb de memoria RAM Monitor VGA o superior.

2.4 Definición de los casos de uso

2.4.1 Definición de los actores

Usuario	Realiza las acciones comunes definidas para todos los que interactúan con la aplicación.
Coordinador de año(Administrador)	Encargado del manejo y configuración de las acciones que se desarrollan en la aplicación. Realiza la gestión de usuarios que intervienen en los procesos de la aplicación. Responsable de evaluar los proyectos educativos y a los actores que interactúan dentro de los mismos, así como velar por que se cumplan los objetivos del año como las actividades programadas en los proyectos.
Profesor guía	Responsable del desarrollo de las acciones que se realizan en el proyecto educativo a nivel de brigada.
Colectivo pedagógico	Visualiza los problemas, los objetivos y acciones presentes en los proyectos educativos.
Estudiante	Visualiza los objetivos y acciones que se desarrollan en el proyecto educativo que le corresponde.

2.5 Diagrama de casos de usos del sistema



2.5.1 Descripción de los casos de uso del sistema

A continuación se muestran las descripciones de los casos de uso del sistema (CUS).

CASO DE USO:	Autenticar usuario
Resumen:	El caso de uso inicia cuando el actor decide interactuar con la aplicación.
Complejidad:	
Prioridad:	
Precondiciones:	
REFERENCIAS	
Actores:	Usuario
Requisitos:	RF. 11
Entidades:	
Casos de uso:	
FLUJO NORMAL DE EVENTOS	
Acción del actor	Respuesta del sistema

	<ol style="list-style-type: none"> 1. El sistema muestra una ventana para la autenticación del actor.
<ol style="list-style-type: none"> 2. El actor ingresa su usuario y contraseña. 	<ol style="list-style-type: none"> 3. El sistema valida que la identificación sea válida. 4. Permite al usuario entrar a la aplicación. 5. El caso de uso termina.

FLUJOS ALTERNOS

Alternativa 1: “En caso que la identificación no sea válida.”

Acción del actor	Respuesta del sistema
	<ol style="list-style-type: none"> 1. Muestra el mensaje de información “Su usuario ó contraseña no es válido.” 2. El sistema regresa al paso 1 del flujo normal de eventos.

CASO DE USO:	Gestionar usuarios
Resumen:	El caso de uso inicia cuando el actor accede a gestionar usuario donde podrá realizar varias acciones, como Insertar, Modificar, Eliminar y Buscar usuario.
Complejidad:	
Prioridad:	
Precondiciones:	
REFERENCIAS	
Actores:	Coordinador de año
Requisitos:	RF. 3, RF. 3.1, RF. 3.2, RF. 3.3, RF. 3.4
Entidades:	
Casos de Uso:	

FLUJO NORMAL DE EVENTOS

Acción del Actor	Respuesta del Sistema
<ol style="list-style-type: none"> 1. El caso de uso se inicia cuando el actor realiza una de las siguientes acciones. <ol style="list-style-type: none"> 1. Dar clic en Insertar usuario. 2. Realizar la acción Buscar usuario. 3. Dar clic en Modificar usuario 4. Dar clic en Eliminar usuario. 	<ol style="list-style-type: none"> 2. Si el actor realiza la acción 1, ver sección Insertar usuario. Si realiza la acción 2, ver sección Modificar usuario. Si realiza la acción 3, ver sección Eliminar usuario.

Sección No.: 1. “Insertar usuario”

Acción del Actor	Respuesta del Sistema
------------------	-----------------------

	<p>1. El sistema muestra la ventana Insertar usuario con los siguientes campos:</p> <ul style="list-style-type: none"> ❖ usuario. ❖ rol.
<p>2. Introduce los datos del usuario y da clic en el botón insertar.</p>	<p>3. El sistema valida que todos los campos estén llenos.</p> <p>4. Guarda los datos insertados y muestra el mensaje “El usuario fue insertado exitosamente.”</p>

FLUJOS ALTERNOS

Alternativa 1: “En la validación alguno de los campos no estén llenos.”

Acción del Actor	Respuesta del Sistema
	<p>1. Muestra el mensaje de información “Verifique los datos seleccionados.”</p>
	<p>2. Regresa al paso 2 del Flujo Normal de Eventos.</p>

Alternativa 2. “Cancelar operación.”

Acción del Actor	Respuesta del Sistema
<p>1. Selecciona la opción de Cancelar operación.</p>	
	<p>2. Regresa a la vista anterior.</p>
	<p>3. El caso de uso termina.</p>

Sección No.: 2. “Modificar usuario”

Acción del Actor	Respuesta del Sistema
<p>7. Selecciona el usuario a modificar.</p>	<p>8. El sistema valida la selección del usuario a modificar.</p>
	<p>9. El sistema muestra la ventana Modificar usuario con los campos correspondientes al profesor seleccionado.</p> <ul style="list-style-type: none"> ❖ usuario. ❖ rol.
<p>10. Modifica los datos del usuario y da clic en el botón aceptar.</p>	<p>11. Guarda los datos insertados y muestra el mensaje “El usuario fue modificado exitosamente.”</p>

FLUJOS ALTERNOS

Alternativa 1: “En la validación para seleccionar el usuario a modificar.”

Acción del Actor	Respuesta del Sistema
------------------	-----------------------

	1. Muestra el mensaje de información “Seleccione el usuario a modificar.”
	2. Regresa al paso 7 del Flujo Normal de Eventos.

Alternativa 2. “Cancelar operación.”

Acción del Actor	Respuesta del Sistema
1. Selecciona la opción de Cancelar operación.	
	2. Regresa a la vista anterior.
	3. El caso de uso termina.

Sección No.: 3. “Eliminar usuario”

Acción del Actor	Respuesta del Sistema
12. Selecciona el usuario a eliminar.	13. El sistema valida la selección del usuario a eliminar.
14. El actor da clic en el botón Sí.	15. El sistema muestra el mensaje de confirmación, “¿Está seguro que desea eliminar el usuario seleccionado?” 16. El sistema elimina el usuario y muestra el mensaje de confirmación “El usuario ha sido eliminado exitosamente.”

FLUJOS ALTERNOS

Alternativa 1: “El actor da clic en el botón No.”

Acción del Actor	Respuesta del Sistema
1. El actor da clic en el botón No	2. El sistema regresa a la vista anterior.

CASO DE USO:	Gestionar profesor
Resumen:	El caso de uso inicia cuando el actor accede a gestionar profesor donde podrá realizar varias acciones, como Insertar, Modificar, Eliminar y Buscar profesor.
Complejidad:	
Prioridad:	
Precondiciones:	
REFERENCIAS	
Actores:	Coordinador de año
Requisitos:	RF.1,RF.1.1,RF.1.2,RF.1.3,RF.1.4
Entidades:	

Casos de Uso:	
FLUJO NORMAL DE EVENTOS	
Acción del Actor	Respuesta del Sistema
<p>1. El caso de uso se inicia cuando el actor realiza una de las siguientes acciones.</p> <ol style="list-style-type: none"> 1. Dar clic en Insertar profesor. 2. Realizarla acción Buscar profesor. 3. Dar clic en Modificar profesor 4. Dar clic en Eliminar profesor. 	<p>2. Si el actor realiza la acción 1, ver sección Insertar profesor. Si realiza la acción 2, ver sección Buscar profesor. Si realiza la acción 3, ver sección Modificar profesor. Si realiza la acción 4, ver sección Eliminar profesor.</p>
Sección No.: 1. “Insertar profesor”	
Acción del Actor	Respuesta del Sistema
	<p>1. El sistema muestra la ventana Insertar Profesor con los siguientes campos:</p> <ul style="list-style-type: none"> ❖ Nombre(s). ❖ Apellidos. ❖ Sexo. ❖ Grupo. ❖ Rol.
<p>2. Introduce los datos del Profesor y da clic en el botón Insertar.</p>	<p>1. El sistema valida que todos los campos estén llenos. 2. Guarda los datos insertados y muestra el mensaje “El profesor fue insertado exitosamente.”</p>
FLUJOS ALTERNOS	
Alternativa 1: “En la validación alguno de los campos no estén llenos.”	
Acción del Actor	Respuesta del Sistema
	<p>1. Muestra el mensaje de información “Verifique los datos seleccionados.”</p>
	<p>2. Regresa al paso 2 del Flujo Normal de Eventos.</p>
Alternativa 2. “Cancelar operación.”	
Acción del Actor	Respuesta del Sistema
<p>1. Selecciona la opción de Cancelar operación.</p>	

	2. Regresa a la vista anterior.
	3. El caso de uso termina.
Sección No.: 2. “Buscar profesor”	
Acción del Actor	Respuesta del Sistema
5. Introduce alguno de los datos correspondientes a la búsqueda: <ul style="list-style-type: none"> ❖ Profesor ❖ Grupo 	6. El sistema muestra el listado de los profesores que cumplen con los criterios de búsqueda.
Sección No.: 3. “Modificar profesor”	
Acción del Actor	Respuesta del Sistema
7. Selecciona el profesor a modificar.	8. El sistema valida la selección del profesor a modificar.
	9. El sistema muestra la ventana Modificar Profesor con los campos correspondientes al profesor seleccionado. <ul style="list-style-type: none"> ❖ Nombre(s). ❖ Apellidos. ❖ Sexo. ❖ Grupo. ❖ Rol.
10. Modifica los datos del profesor y da clic en el botón aceptar.	11. Guarda los datos modificados y muestra el mensaje “El profesor fue modificado exitosamente.”
FLUJOS ALTERNOS	
Alternativa 1: “En la validación para seleccionar el objetivo a modificar.”	
Acción del Actor	Respuesta del Sistema
	1. Muestra el mensaje de información “Seleccione el profesor a modificar.”
	2. Regresa al paso 7 del Flujo Normal de Eventos.
Alternativa 2: “Cancelar operación.”	
Acción del Actor	Respuesta del Sistema
1. Selecciona la opción de Cancelar operación.	

	2. Regresa a la vista anterior.
	3. El caso de uso termina.
Sección No.: 4. “Eliminar profesor”	
Acción del Actor	Respuesta del Sistema
12. Selecciona el profesor a eliminar.	13. El sistema valida la selección del profesor a eliminar.
14. El actor da clic en el botón Sí.	15. El sistema muestra el mensaje de confirmación, “¿Está seguro que desea eliminar el profesor seleccionado?” 16. El sistema elimina el profesor y muestra el mensaje de confirmación “El profesor ha sido eliminado exitosamente.”
FLUJOS ALTERNOS	
Alternativa 1: “El actor da clic en el botón No.”	
Acción del Actor	Respuesta del Sistema
1. El actor da clic en el botón No	2. El sistema regresa a la vista anterior Buscar Profesor.

CASO DE USO:	Gestionar grupo
Resumen:	El caso de uso inicia cuando el actor accede a gestionar los grupos donde podrá realizar varias acciones, como Insertar, Modificar, Eliminar y Buscar grupo.
Complejidad:	
Prioridad:	
Precondiciones:	
REFERENCIAS	
Actores:	Coordinador de año
Requisitos:	RF. 2, RF. 2.1, RF. 2.2, RF. 2.3, RF. 2.4
Entidades:	
Casos de Uso:	
FLUJO NORMAL DE EVENTOS	
Acción del Actor	Respuesta del Sistema
1. El caso de uso se inicia cuando el actor realiza una de las siguientes acciones. <ul style="list-style-type: none"> 1. Dar clic en Insertar grupo. 2. Dar clic en Modificar uno de los grupos. 	2. Si el actor realiza la acción 1, ver sección Insertar grupo. Si realiza la acción 2, ver sección Modificar grupo. Si realiza la acción 3, ver sección Eliminar grupo. Si realiza la acción 4, ver sección Buscar grupo.

<ol style="list-style-type: none"> 3. Dar clic en Eliminar uno de los grupos. 4. Realiza otra de las acciones del escenario Buscar grupo 	
Sección No.: 1. “Insertar grupo”	
Acción del Actor	Respuesta del Sistema
	<ol style="list-style-type: none"> 1. El sistema muestra la ventana Insertar Grupo con los siguientes campos: <ul style="list-style-type: none"> ❖ Grupo. ❖ Profesor guía. ❖ Matrícula. ❖ Coordinador de año.
<ol style="list-style-type: none"> 2. Introduce los datos del grupo y da clic en el botón aceptar. 	<ol style="list-style-type: none"> 3. El sistema valida que todos los campos estén llenos. 4. Guarda los datos insertados y muestra el mensaje “El grupo fue insertado exitosamente.”
FLUJOS ALTERNOS	
Alternativa 1: “En la validación alguno de los campos no estén llenos.”	
Acción del Actor	Respuesta del Sistema
	<ol style="list-style-type: none"> 1. Muestra el mensaje de información “Verifique los datos seleccionados.”
	<ol style="list-style-type: none"> 2. Regresa al paso 2 del Flujo Normal de Eventos.
Alternativa 2. “Cancelar operación.”	
Acción del Actor	Respuesta del Sistema
<ol style="list-style-type: none"> 1. Selecciona la opción de Cancelar operación. 	
	<ol style="list-style-type: none"> 2. Regresa a la vista anterior.
	<ol style="list-style-type: none"> 3. El caso de uso termina.
Sección No.: 2. “Buscar grupo”	
Acción del Actor	Respuesta del Sistema
<ol style="list-style-type: none"> 5. Introduce alguno de los datos correspondientes a la búsqueda: <ul style="list-style-type: none"> ❖ Grupo 	<ol style="list-style-type: none"> 6. El sistema muestra el listado de los grupos que se encuentran registrados en la aplicación.
Sección No.: 3. “Modificar grupo”	

Capítulo 2: Características del Sistema

Acción del Actor	Respuesta del Sistema
	<p>7. El sistema muestra la ventana Modificar grupo con los campos correspondientes al objetivo seleccionado.</p> <ul style="list-style-type: none"> ❖ Grupo. ❖ Profesor Guía. ❖ Matrícula. ❖ Coordinador de año.
8. Modifica los datos del grupo y da clic en el botón aceptar.	<p>9. El sistema valida que todos los campos estén llenos.</p> <p>10. Guarda los datos insertados y muestra el mensaje “Su grupo fue modificado exitosamente.”</p>
FLUJOS ALTERNOS	
Alternativa 1: “En la validación alguno de los campos no estén llenos.”	
Acción del Actor	Respuesta del Sistema
	1. Muestra el mensaje de información “Verifique los datos seleccionados.”
	2. Regresa al paso 5 del Flujo Normal de Eventos.
Alternativa 2. “Cancelar operación.”	
Acción del Actor	Respuesta del Sistema
1. Selecciona la opción de Cancelar operación.	
	2. Regresa a la vista anterior.
	3. El caso de uso termina.
Sección No.: 4. “Eliminar grupo”	
Acción del Actor	Respuesta del Sistema
11. El actor da clic en el botón Sí.	<p>12. El sistema muestra el mensaje de confirmación, “¿Está seguro que desea eliminar el grupo seleccionado?”</p> <p>13. El sistema elimina el grupo y muestra el mensaje de confirmación “El grupo ha sido eliminado exitosamente.”</p>
FLUJOS ALTERNOS	
Alternativa 1: “El actor da clic en el botón No.”	
Acción del Actor	Respuesta del Sistema
1. El actor da clic en el botón No	2. El sistema regresa a la vista anterior Buscar grupo.

CASO DE USO:	Gestionar problema	
Resumen:	El caso de uso inicia cuando el actor accede a gestionar los problemas donde podrá realizar varias acciones, como Insertar, Modificar, Eliminar y Buscar problema.	
Complejidad:		
Prioridad:		
Precondiciones:		
REFERENCIAS		
Actores:	Profesor guía	
Requisitos:	RF. 6, RF. 6.1, RF. 6.2, RF. 6.3, RF. 6.4,	
Entidades:		
Casos de uso:		
FLUJO NORMAL DE EVENTOS		
Acción del actor	Respuesta del sistema	
<p>1. El caso de uso se inicia cuando el actor realiza una de las siguientes acciones.</p> <ol style="list-style-type: none"> 1. Dar clic en Insertar problema. 2. Realiza la acción Buscar problema. 3. Dar clic en Modificar problema. 4. Dar clic en Eliminar problema. 	<p>2. Si el actor realiza la acción 1, ver sección Insertar problema. Si realiza la acción 2, ver sección Buscar problema. Si realiza la acción 3, ver sección Modificar problema. Si realiza la acción 4, ver sección Eliminar problema.</p>	
Sección No.: 1. “Insertar problema”		
Acción del actor	Respuesta del sistema	
	<p>1. El sistema muestra la ventana Insertar problema con los siguientes campos:</p> <ul style="list-style-type: none"> ❖ No problema ❖ Dimensión ❖ Grupo. ❖ problema. ❖ Estado 	
<p>2. Introduce los datos del problema y da clic en el botón insertar.</p>	<p>3. El sistema valida que todos los campos estén llenos. 4. Guarda los datos insertados y muestra el mensaje “Su problema fue insertado exitosamente.”</p>	
FLUJOS ALTERNOS		
Alternativa 1: “En la validación alguno de los campos no estén llenos.”		
Acción del actor	Respuesta del sistema	
	<p>1. Muestra el mensaje de información “Verifique los datos seleccionados.”</p>	
	<p>2. Regresa al paso 2 del flujo normal de eventos.</p>	

Alternativa 2. “Cancelar operación.”

Acción del actor	Respuesta del sistema
1. Selecciona la opción de cancelar operación.	
	2. Regresa a la vista anterior.
	3. El caso de uso termina.

Sección No.: 2. “Buscar problema”

Acción del actor	Respuesta del sistema
5. Introduce alguno de los datos correspondientes a la búsqueda: <ul style="list-style-type: none"> ❖ No problema ❖ Dimensión ❖ Grupo. ❖ problema. 	6. El sistema muestra el listado de los problemas que cumplen con los criterios de búsqueda.

Sección No.: 3. “Modificar problema”

Acción del actor	Respuesta del sistema
7. Selecciona el problema a modificar.	8. El sistema valida la selección del problema a modificar.
	9. El sistema muestra la ventana modificar problema con los campos correspondientes al problema seleccionado. <ul style="list-style-type: none"> ❖ No problema ❖ Dimensión ❖ Grupo. ❖ problema. ❖ Estado
10. Modifica los datos del problema y da clic en el botón aceptar.	11. Guarda los datos modificados y muestra el mensaje “Su problema fue modificado exitosamente.”

FLUJOS ALTERNOS

Alternativa 1: “En la validación para seleccionar el problema a modificar.”

Acción del actor	Respuesta del sistema
	1. Muestra el mensaje de información “Seleccione el problema a modificar.”
	2. Regresa al paso 7 del flujo normal de eventos.

Alternativa 2. “Cancelar operación.”

Acción del actor	Respuesta del sistema
1. Selecciona la opción de cancelar operación.	
	2. Regresa a la vista anterior.
	3. El caso de uso termina.

Sección No.: 4. “Eliminar problema”

Acción del actor	Respuesta del sistema
12. Selecciona el problema a eliminar.	13. El sistema valida la selección del problema a eliminar.
14. El actor da clic en el botón Sí.	15. El sistema muestra el mensaje de confirmación, “¿Está seguro que desea eliminar el problema seleccionado?” 16. El sistema elimina el problema y muestra el mensaje de confirmación “El problema ha sido eliminado exitosamente.”

FLUJOS ALTERNOS

Alternativa 1: “En la validación para seleccionar el problema a eliminar.”

Acción del actor	Respuesta del sistema
	1. Muestra el mensaje de información “Seleccione el problema a eliminar.” 2. Regresa al paso 12 del flujo normal de eventos.

Alternativa 2. “El actor da clic en el botón No.”

Acción del actor	Respuesta del sistema
1. El actor da clic en el botón No	2. El sistema regresa a la vista anterior Buscar problema.

CASO DE USO:	Gestionar objetivo
Resumen:	El caso de uso inicia cuando el actor accede a gestionar los objetivos donde podrá realizar varias acciones, como Insertar, Modificar, Eliminar y Buscar objetivo.
Complejidad:	
Prioridad:	
Precondiciones:	
REFERENCIAS	
Actores:	Profesor guía
Requisitos:	RF. 4, RF. 4.1, RF. 4.2, RF. 4.3, RF. 4.4

Entidades:	
Casos de uso:	
FLUJO NORMAL DE EVENTOS	
Acción del actor	Respuesta del sistema
<p>1. El caso de uso se inicia cuando el actor realiza una de las siguientes acciones.</p> <p style="padding-left: 40px;">5. Dar clic en Insertar objetivo.</p> <p style="padding-left: 40px;">6. Realiza la acción Buscar objetivo.</p> <p style="padding-left: 40px;">7. Dar clic en Modificar objetivo.</p> <p style="padding-left: 40px;">8. Dar clic en Eliminar objetivo.</p>	<p>2. Si el actor realiza la acción 1, ver sección Insertar objetivo. Si realiza la acción 2, ver sección Buscar objetivo. Si realiza la acción 3, ver sección Modificar objetivo. Si realiza la acción 4, ver sección Eliminar objetivo.</p>
Sección No.: 1. “Insertar objetivo”	
Acción del actor	Respuesta del sistema
	<p>1. El sistema muestra la ventana Insertar objetivo con los siguientes campos:</p> <ul style="list-style-type: none"> ❖ No objetivo ❖ Dimensión ❖ Grupo. ❖ Objetivo. ❖ Estado
<p>2. Introduce los datos del objetivo y da clic en el botón insertar.</p>	<p>3. El sistema valida que todos los campos estén llenos.</p> <p>4. Guarda los datos insertados y muestra el mensaje “Su objetivo fue insertado exitosamente.”</p>
FLUJOS ALTERNOS	
Alternativa 1: “En la validación alguno de los campos no estén llenos.”	
Acción del actor	Respuesta del sistema
	<p>1. Muestra el mensaje de información “Verifique los datos seleccionados.”</p>
	<p>2. Regresa al paso 2 del flujo normal de eventos.</p>
Alternativa 2. “Cancelar operación.”	
Acción del actor	Respuesta del sistema
<p>1. Selecciona la opción de cancelar operación.</p>	
	<p>2. Regresa a la vista anterior.</p>
	<p>3. El caso de uso termina.</p>

Sección No.: 2. “Buscar objetivo”	
Acción del actor	Respuesta del sistema
5. Introduce alguno de los datos correspondientes a la búsqueda: <ul style="list-style-type: none"> ❖ No objetivo ❖ Dimensión ❖ Grupo. ❖ Objetivo. 	6. El sistema muestra el listado de los objetivos que cumplen con los criterios de búsqueda.
Sección No.: 3. “Modificar objetivo”	
Acción del actor	Respuesta del sistema
7. Selecciona el objetivo a modificar.	8. El sistema valida la selección del objetivo a modificar.
	9. El sistema muestra la ventana modificar objetivo con los campos correspondientes al objetivo seleccionado. <ul style="list-style-type: none"> ❖ No Objetivo ❖ Dimensión ❖ Grupo. ❖ Objetivo. ❖ Estado
10. Modifica los datos del objetivo y da clic en el botón aceptar.	11. Guarda los datos modificados y muestra el mensaje “Su objetivo fue modificado exitosamente.”
FLUJOS ALTERNOS	
Alternativa 1: “En la validación para seleccionar el objetivo a modificar.”	
Acción del actor	Respuesta del sistema
	1. Muestra el mensaje de información “Seleccione el objetivo a modificar.”
	2. Regresa al paso 7 del flujo normal de eventos.
Alternativa 2. “Cancelar operación.”	
Acción del actor	Respuesta del sistema
4. Selecciona la opción de cancelar operación.	
	5. Regresa a la vista anterior.
	6. El caso de uso termina.

Sección No.: 4. “Eliminar objetivo”	
Acción del actor	Respuesta del sistema
12. Selecciona el objetivo a eliminar.	13. El sistema valida la selección del objetivo a eliminar.
14. El actor da clic en el botón Sí.	15. El sistema muestra el mensaje de confirmación, “¿Está seguro que desea eliminar el objetivo seleccionado?” 16. El sistema elimina el objetivo y muestra el mensaje de confirmación “El objetivo ha sido eliminado exitosamente.”
FLUJOS ALTERNOS	
Alternativa 1: “En la validación para seleccionar el objetivo a eliminar.”	
Acción del actor	Respuesta del sistema
	1. Muestra el mensaje de información “Seleccione el objetivo a eliminar.” 2. Regresa al paso 12 del flujo normal de eventos.
Alternativa 2. “El actor da clic en el botón No.”	
Acción del actor	Respuesta del sistema
1. El actor da clic en el botón No	2. El sistema regresa a la vista anterior Buscar objetivo.

CASO DE USO:	Gestionar acción
Resumen:	El caso de uso inicia cuando el actor accede a gestionar las acciones del proyecto educativo donde podrá Insertar, Modificar, Eliminar y Buscar la Acción.
Complejidad:	
Prioridad:	
Precondiciones:	
REFERENCIAS	
Actores:	Profesor guía
Requisitos:	RF. 5, RF. 5.1, RF .5.2, RF .5.3, RF. 5.4, RF.5.5
Entidades:	
Casos de Uso:	
FLUJO NORMAL DE EVENTOS	
Acción del Actor	Respuesta del Sistema
1. El caso de uso se inicia cuando el actor realiza una de las siguientes acciones. 9. Dar clic en Insertar acción. 10. Realizar la acción Buscar	2. Si el actor realiza la acción 1, ver sección Insertar acción. Si realiza la acción 2, ver sección Buscar acción. Si realiza la acción 3, ver sección Modificar acción. Si realiza la acción 4, ver sección Eliminar acción. Si realiza la acción 5, ver sección Reporte acción.

acción. 11. Dar clic en Modificar acción. 12. Dar clic en Eliminar acción. 13. Dar clic en Reporte acción.	
Sección No.: 1. “Insertar acción”	
Acción del Actor	Respuesta del Sistema
	1. El sistema muestra la ventana Insertar Acción con los siguientes campos: <ul style="list-style-type: none"> ❖ No Acción. ❖ Objetivo. ❖ Dimensión. ❖ Criterio de Medida. ❖ Grupo. ❖ Valores Relacionados. ❖ Descripción de la Acción. ❖ Fecha de Cumplimiento. ❖ Responsable(s) ❖ Participantes.
2. Introduce los datos de la acción y da clic en el botón insertar.	3. El sistema valida que todos los campos estén llenos. 4. Guarda los datos insertados y muestra el mensaje “Su acción fue insertada exitosamente.”
FLUJOS ALTERNOS	
Alternativa 1: “En la validación alguno de los campos no estén llenos.”	
Acción del Actor	Respuesta del Sistema
	1. Muestra el mensaje de información “Verifique los datos seleccionados.”
	2. Regresa al paso 2 del Flujo Normal de Eventos.
Alternativa 2. “Cancelar operación.”	
Acción del Actor	Respuesta del Sistema
1. Selecciona la opción de Cancelar operación.	
	2. Regresa a la vista anterior.
	3. El caso de uso termina.

Sección No.: 2. “Buscar acción”	
Acción del Actor	Respuesta del Sistema
<p>5. Introduce alguno de los datos correspondientes a la búsqueda:</p> <ul style="list-style-type: none"> ❖ Grupo ❖ No Acción. ❖ Objetivo. ❖ Fecha de Cumplimiento. ❖ Responsable. ❖ Dimensión. 	<p>6. El sistema muestra el listado de las acciones que cumplen con los criterios de búsqueda.</p>
Sección No.: 3. “Modificar acción”	
Acción del Actor	Respuesta del Sistema
<p>7. Selecciona la acción a modificar.</p>	<p>8. El sistema valida la selección de la acción a modificar.</p>
	<p>9. El sistema muestra la ventana Modificar Acción con los campos correspondientes a la acción seleccionada.</p> <ul style="list-style-type: none"> ❖ No Acción. ❖ Objetivo. ❖ Dimensión. ❖ Criterio de Medida. ❖ Grupo. ❖ Valores Relacionados. ❖ Descripción de la Acción. ❖ Fecha de Cumplimiento. ❖ Responsable(s) ❖ Participantes.
<p>10. Modifica los datos de la acción y da clic en el botón aceptar.</p>	<p>11. Guarda los datos modificados y muestra el mensaje “Su acción fue modificada exitosamente.”</p>
FLUJOS ALTERNOS	
Alternativa 1: “En la validación para seleccionar la acción a modificar.”	
Acción del Actor	Respuesta del Sistema
	<p>1. Muestra el mensaje de información “Seleccione la acción a modificar.”</p>

	2. Regresa al paso 7 del Flujo Normal de Eventos.
Alternativa 2. “Cancelar operación.”	
Acción del Actor	Respuesta del Sistema
1. Selecciona la opción de Cancelar operación.	
	2. Regresa a la vista anterior.
	3. El caso de uso termina.
Sección No.: 4. “Eliminar acción”	
Acción del Actor	Respuesta del Sistema
12. Selecciona la acción a eliminar.	13. El sistema valida la selección de la acción a eliminar.
14. El actor da clic en el botón Sí.	15. El sistema muestra el mensaje de confirmación, “¿Está seguro que desea eliminar la acción seleccionada?” 16. El sistema elimina la acción y muestra el mensaje de confirmación “La acción ha sido eliminada exitosamente.”
FLUJOS ALTERNOS	
Alternativa 1: “En la validación para seleccionar la acción a eliminar.”	
Acción del Actor	Respuesta del Sistema
	1. Muestra el mensaje de información “Seleccione la acción a eliminar.” 2. Regresa al paso 12 del Flujo Normal de Eventos.
Alternativa 2. “El actor da clic en el botón No.”	
Acción del Actor	Respuesta del Sistema
1. El actor da clic en el botón No	2. El sistema regresa a la vista anterior Buscar acción.
Sección No.: 5. “Reporte acción”	
Acción del Actor	Respuesta del Sistema
17. Selecciona la acción Reporte.	18. El sistema genera un documento con las acciones existentes. 19. El caso de uso termina.

CASO DE USO:	Visualizar objetivos	
Resumen:	El caso de uso inicia cuando el actor accede a la acción visualizar objetivos, ya sea según el criterio de búsqueda, seleccionando el grupo del cual desea visualizar los objetivos ó según al grupo donde pertenezca.	
Complejidad:		
Prioridad:		
Precondiciones:		
REFERENCIAS		
Actores:	Colectivo Pedagógico, Estudiante	
Requisitos:	RF. 8	
Entidades:		
Casos de Uso:		
FLUJO NORMAL DE EVENTOS		
Acción del Actor	Respuesta del Sistema	
1. El caso de uso se inicia cuando el actor entra a la funcionalidad Visualizar objetivos.	2. Si el actor registrado en la aplicación pertenece al colectivo pedagógico, ver sección buscar grupo. Si el actor registrado en la aplicación es un estudiante, ver sección 2, visualizar objetivos.	
Sección No.: 1. “Buscar grupo ”		
Acción del Actor	Respuesta del Sistema	
1. El usuario selecciona el grupo, del cual desea ver los objetivos.	2. El sistema muestra los objetivos que presenta el grupo seleccionado.	
FLUJOS ALTERNOS		
Alternativa 1. “Cancelar operación.”		
Acción del Actor	Respuesta del Sistema	
1. Selecciona la opción de Cancelar operación.		
	2. Regresa a la vista anterior.	
	3. El caso de uso termina.	
Sección No.: 2. “Visualizar objetivos”		
Acción del Actor	Respuesta del Sistema	
	3. El sistema muestra el listado de los objetivos del grupo, al cual pertenece el estudiante que se registro en la aplicación.	
	4. El caso de uso termina.	

CASO DE USO:	Visualizar problemas
Resumen:	El caso de uso inicia cuando el actor accede a la acción visualizar problemas, ya sea

	según el criterio de búsqueda, seleccionando el grupo del cual desea visualizar los problemas ó según al grupo donde pertenezca.
Complejidad:	
Prioridad:	
Precondiciones:	
REFERENCIAS	
Actores:	Colectivo pedagógico, Estudiante
Requisitos:	RF. 10
Entidades:	
Casos de Uso:	
FLUJO NORMAL DE EVENTOS	
Acción del Actor	Respuesta del Sistema
1. El caso de uso se inicia cuando el actor entra a la funcionalidad Visualizar problemas.	2. Si el actor registrado en la aplicación pertenece al colectivo pedagógico, ver sección buscar problema. Si el actor registrado en la aplicación es un estudiante, ver sección 2, visualizar objetivos.
Sección No.: 1. “Buscar grupo ”	
Acción del Actor	Respuesta del Sistema
1. El usuario selecciona el grupo, del cual desea ver los problemas.	2. El sistema muestra los problemas que presenta el grupo seleccionado.
FLUJOS ALTERNOS	
Alternativa 1. “Cancelar operación.”	
Acción del Actor	Respuesta del Sistema
1. Selecciona la opción de Cancelar operación.	
	2. Regresa a la vista anterior.
	3. El caso de uso termina.
Sección No.: 2. “Visualizar problemas”	
Acción del Actor	Respuesta del Sistema
	3. El sistema muestra el listado de los problemas del grupo, al cual pertenece el estudiante que se registro en la aplicación.
	4. El caso de uso termina.
CASO DE USO:	Visualizar acciones
Resumen:	El caso de uso inicia cuando el actor accede a la acción Visualizar acciones, ya sea según el criterio de búsqueda, seleccionando el grupo del cual desea visualizar las acciones ó según al grupo donde pertenezca.
Complejidad:	
Prioridad:	

Precondiciones:	
REFERENCIAS	
Actores:	Colectivo pedagógico, Estudiante
Requisitos:	RF. 9
Entidades:	
Casos de Uso:	
FLUJO NORMAL DE EVENTOS	
Acción del Actor	Respuesta del Sistema
1. El caso de uso se inicia cuando el actor entra a la funcionalidad Visualizar acciones.	2. Si el actor registrado en la aplicación pertenece al colectivo pedagógico, ver sección Buscar acciones. Si el actor registrado en la aplicación es un estudiante, ver sección 2, visualizar acciones.
Sección No.: 1. “Buscar acciones ”	
Acción del Actor	Respuesta del Sistema
1. El usuario selecciona el grupo, del cual desea ver las acciones.	2. El sistema muestra las acciones que presenta el grupo seleccionado.
FLUJOS ALTERNOS	
Alternativa 1. “Cancelar operación.”	
Acción del Actor	Respuesta del Sistema
1. Selecciona la opción de Cancelar operación.	
	2. Regresa a la vista anterior.
	3. El caso de uso termina.
Sección No.: 2. “Visualizar acciones”	
Acción del Actor	Respuesta del Sistema
	3. El sistema muestra el listado de los acciones del grupo, al cual pertenece el estudiante que se registro en la aplicación.
	4. El caso de uso termina.

Conclusiones

En este capítulo se realizó un estudio de la lógica del negocio a través del modelo de dominio. Se identificaron los requerimientos a cumplir por parte del sistema, tanto funcional como no funcional. Se definieron los casos de usos del sistema donde se encuentran las principales funcionalidades a implementar, además se describió cómo es la interacción entre los usuarios y los casos de usos del sistema, mediante el diagrama y la descripción de los mismos.

Capítulo 3: Diseño del sistema

Introducción

En este capítulo se desarrollan los diagramas de clases tanto del análisis como del diseño que participan en la realización de los casos de usos arquitectónicamente significativos. Se representa distintos diagramas de secuencia de dichos casos de usos. Se muestra una breve descripción de las clases entidades y controladoras empleadas y de las tablas de la base de datos, de la que se muestra a su vez los diagramas entidad – relación.

3.1 Flujo de Trabajo de análisis y diseño.

El objetivo de esta disciplina es transformar los requerimientos en un diseño de cómo va hacer implementado el sistema, evolucionar hacia una arquitectura de software robusta y adaptar el diseño para que coincida con el ambiente de implementación, diseñando el sistema con un enfoque hacia el rendimiento. El análisis consiste en obtener una visión que se preocupa de ver qué hace el sistema de software a desarrollar, por lo cual se interesa solo en los requerimientos funcionales. Mientras que el diseño es un refinamiento del análisis que toma en cuenta los requerimientos no funcionales, por lo cual se centra en cómo el sistema cumple sus objetivos. [27]

3.2 Modelo de análisis.

Este modelo se utiliza con el fin de estructurar el sistema a implementar, sirve como la abstracción del modelo de diseño. El objetivo de hacerlo es conseguir una comprensión más precisa de los requisitos definiendo conceptos claves que describen el sistema. Este artefacto es opcional, pero también tiene a su vez la propiedad de ser temporal. Su utilidad radica en que permite un acercamiento visual del sistema. [28]

3.2.1 Diagrama de clases del análisis

El diagrama de clases del análisis es un artefacto en el que se representa los conceptos en un dominio del problema. Representa las cosas del mundo real es un diagrama estático que describe la estructura de un sistema mostrando sus clases, atributos y las relaciones entre ellos. Está compuesto por clases y sus relaciones. Las clases del análisis se centran en los requerimientos funcionales, representan conceptos y relaciones del dominio. Estas poseen atributos y entre ellas pueden existir relaciones de asociación, agregación / composición, generalización / especialización. [29]

RUP propone varias clasificaciones para estas clases:

- ❖ **Clase Interfaz (CI):** modela la interacción entre el sistema y sus actores.
- ❖ **Clase Controladora (CC):** coordina la realización de uno o unos pocos casos de uso coordinando las actividades de los objetos que implementan la funcionalidad del caso de uso.
- ❖ **Clase Entidad (CE):** modela información que posee larga vida y es a menudo persistente.

La herramienta CASE Visual Paradigm, representa los tipos de clases a los que se ha hecho referencia de la siguiente forma:



Figura 1: Representación de los tipos de clases del análisis.

A continuación se muestran los diagramas de clases del análisis de los casos de uso del sistema.

Diagrama de clases del análisis del CUS Gestionar usuario

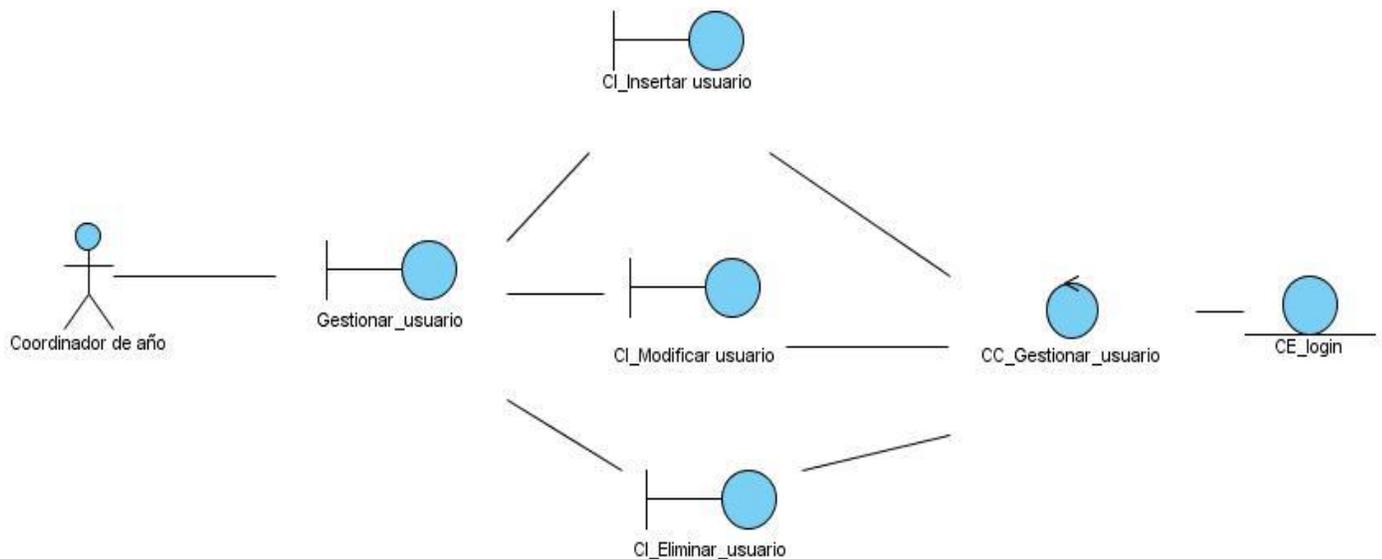


Diagrama de clases del análisis del CUS Gestionar profesor

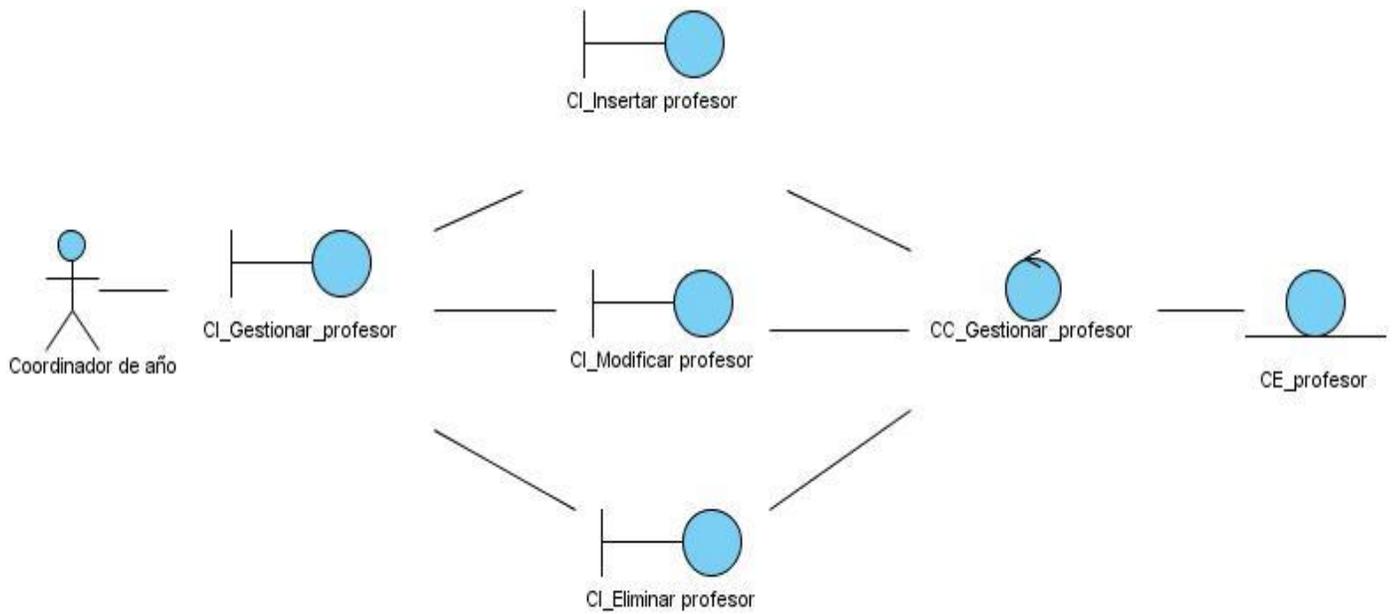


Diagrama de clases del análisis del CUS Gestionar grupo

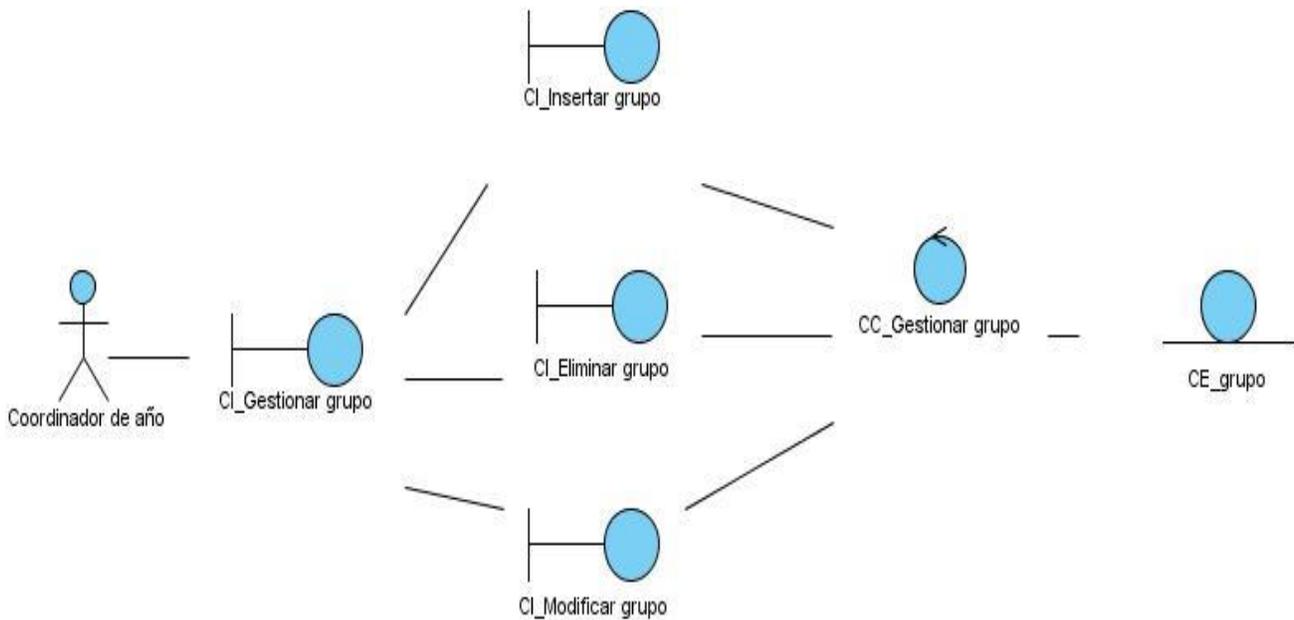


Diagrama de clases del análisis del CUS Gestionar objetivo

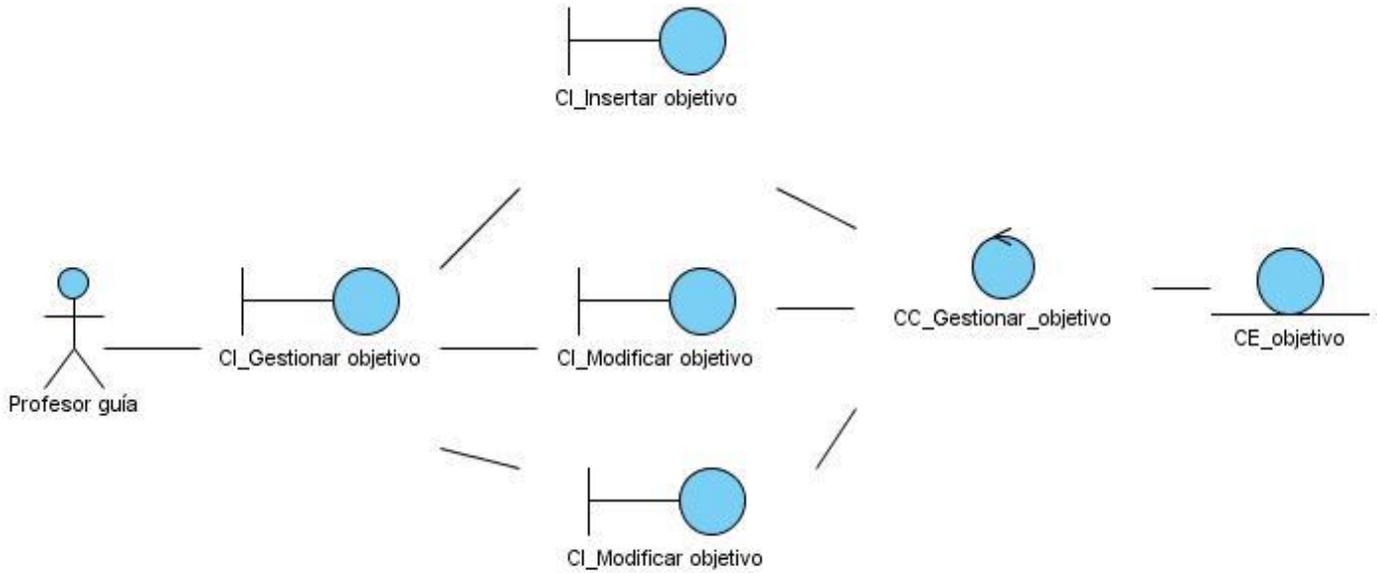


Diagrama de clases del análisis del CUS Gestionar acción

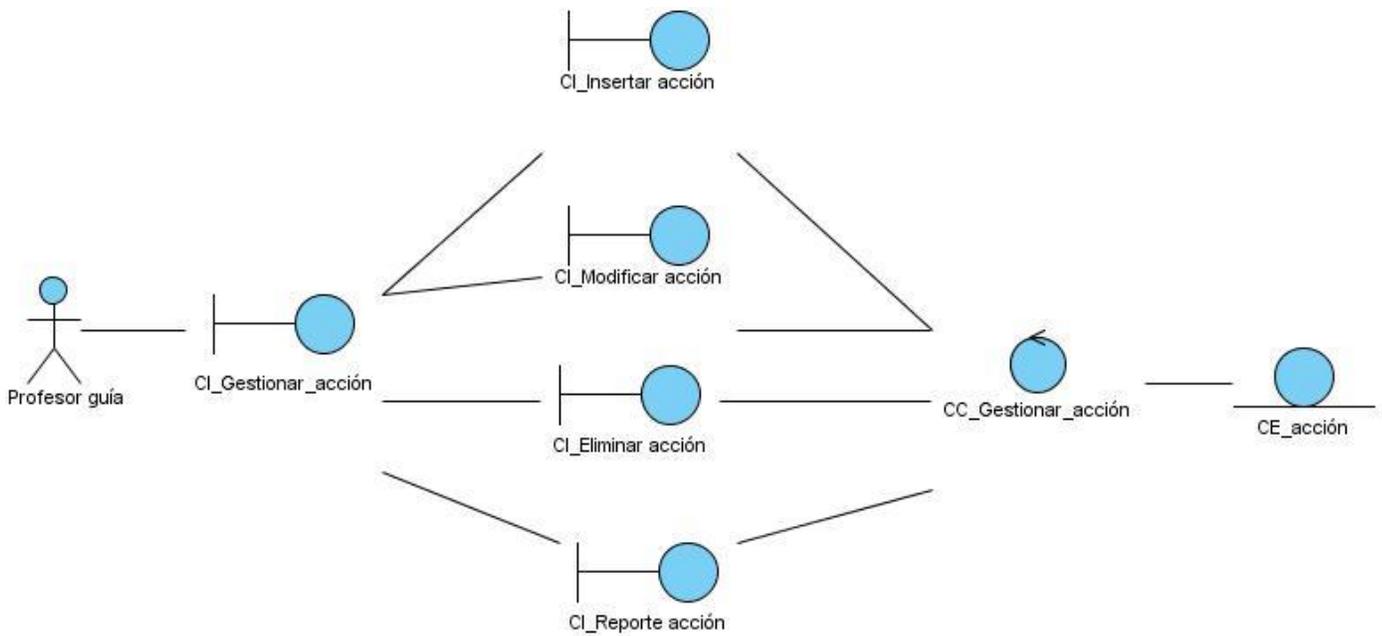


Diagrama de clases del análisis del CUS Gestionar problema

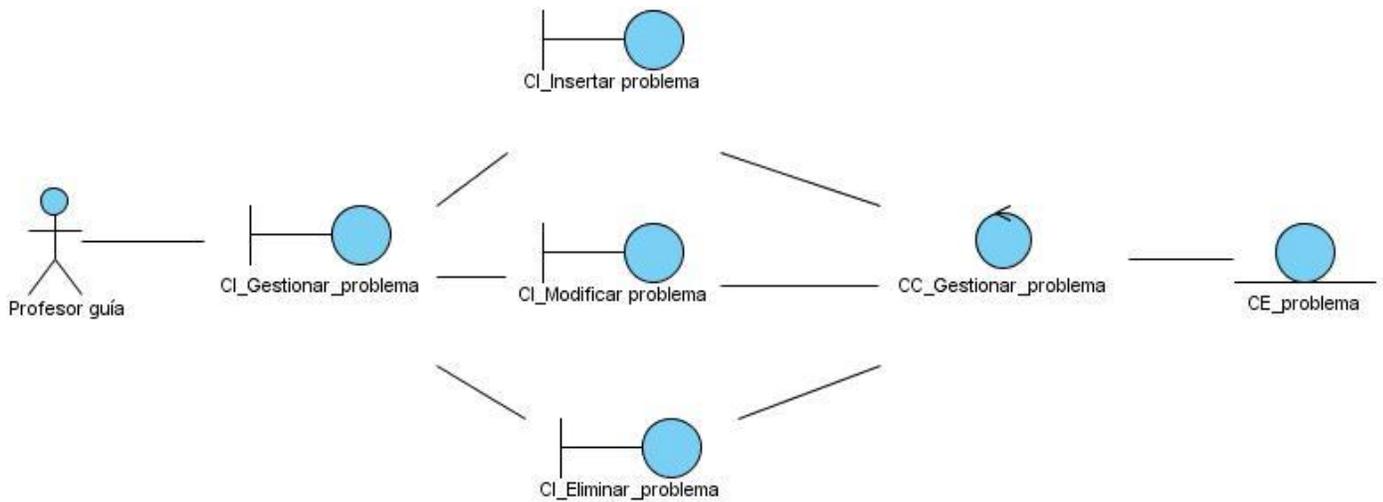


Diagrama de clases del análisis del CUS Visualizar objetivo

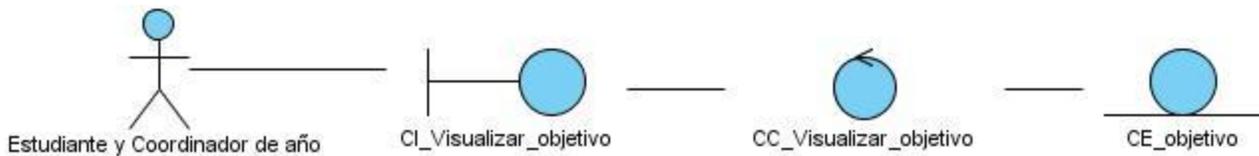


Diagrama de clases del análisis del CUS Visualizar problema

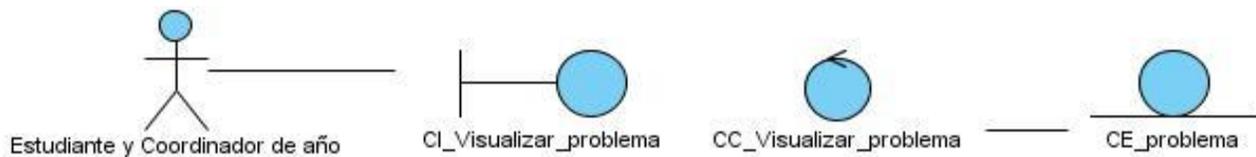
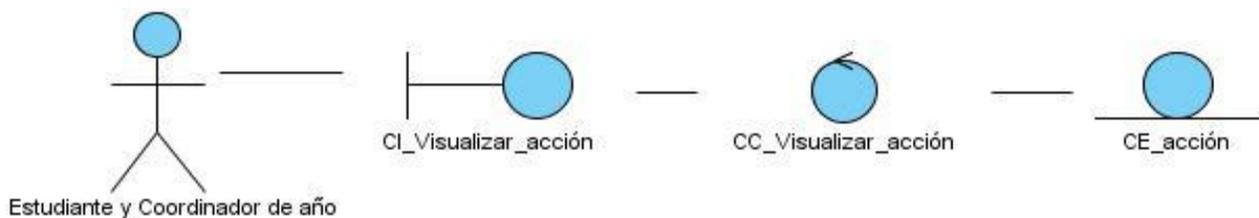


Diagrama de clases del análisis del CUS Visualizar acción



3.3 Modelo de diseño

El modelo de diseño es un proceso para la definición detallada de un sistema con el fin de la realización física de los casos de uso para cubrir las funciones que realizará el sistema y otras restricciones del entorno de implementación que tienen impacto en el mismo, por tanto, en él se definen las clases del diseño que conformarán el sistema a implementar con la arquitectura anteriormente estudiada, modelo vista controlador.

3.3.1 Clases del diseño

- ❖ **Páginas servidoras:** son las encargadas de la construcción de forma dinámica de las páginas clientes y sirven de enlace entre estas y el resto de las clases.
- ❖ **Páginas controladoras:** son las responsables de realizar las operaciones que responden a los procesos de negocio y dar respuestas a las solicitudes hechas por el usuario.
- ❖ **Páginas clientes:** son las páginas encargadas de permitir a los usuarios interactuar con el sistema tanto para hacer solicitudes como para que sean mostradas las respuestas a las mismas.
- ❖ **Clases entidad:** son las responsables de la persistencia de los datos físicamente.



3.3.2 Patrones de diseño

Un patrón de diseño es una descripción de un problema y su solución. Estos evitan la reiteración en la búsqueda de soluciones a problemas ya conocidos y solucionados. Muchos patrones ofrecen además orientación sobre como asignar las responsabilidades a los objetos ante determinada categoría de problemas. Estos son los llamados patrones GRASP, siglas de Patrones Generales de Software para Asignación de Responsabilidades, aunque se considera que más que patrones propiamente dichos, son una serie de "buenas prácticas" de aplicación recomendable en el diseño de software.

De este tipo de patrones serán usados durante el modelado del diseño: el experto, Creador, Controlador, Alta Cohesión y Bajo Acoplamiento. [31]

Experto: indica que la responsabilidad de la creación de un objeto siempre se debe asignar al experto en información, es decir, la responsabilidad recae sobre la clase que conoce toda la información necesaria para poder crearlo. El grasp de experto en información es el principio básico de asignación de responsabilidades.

Beneficios: se mantiene el encapsulamiento, los objetos utilizan su propia información para llevar a cabo sus tareas. Se distribuye el comportamiento entre las clases que contienen la información requerida. Son más fáciles de entender y mantener.

Creador: el patrón creador ayuda a identificar quién debe ser el responsable de la creación (o instanciación) de nuevos objetos o clases.

La nueva instancia deberá ser creada por la clase que:

- ❖ Tiene la información necesaria para realizar la creación del objeto.
- ❖ Usa directamente las instancias creadas del objeto.
- ❖ Almacena o maneja varias instancias de la clase.

Controlador: sirve como intermediario entre una determinada interfaz y el algoritmo que la implementa, de tal forma que es la que recibe los datos del usuario y la que los envía a las distintas clases según el método llamado. Este patrón sugiere que la lógica de negocios debe estar separada de la capa de presentación, esto para aumentar la reutilización de código y a la vez tener un mayor control. Se recomienda dividir los eventos del sistema en el mayor número de controladores para poder aumentar la cohesión y disminuir el acoplamiento.

Alta cohesión y bajo acoplamiento: se puede separar, aunque están íntimamente ligados, de hecho, si se esfuerza en aumentar mucho la cohesión del software, es muy posible que se perjudique el acoplamiento aumentándolo, y por el contrario si se reduce mucho el acoplamiento, se verá disminuida la cohesión:

Alta cohesión: la información que almacena una clase debe de ser coherente y estar en la mayor medida de lo posible relacionada con la clase.

Bajo acoplamiento: es la idea de tener las clases lo menos ligadas entre sí que se pueda. De tal forma que en caso de producirse una modificación en alguna de ellas, se tenga la mínima repercusión posible en el resto de las clases, potenciando la reutilización, y disminuyendo la dependencia entre las clases.

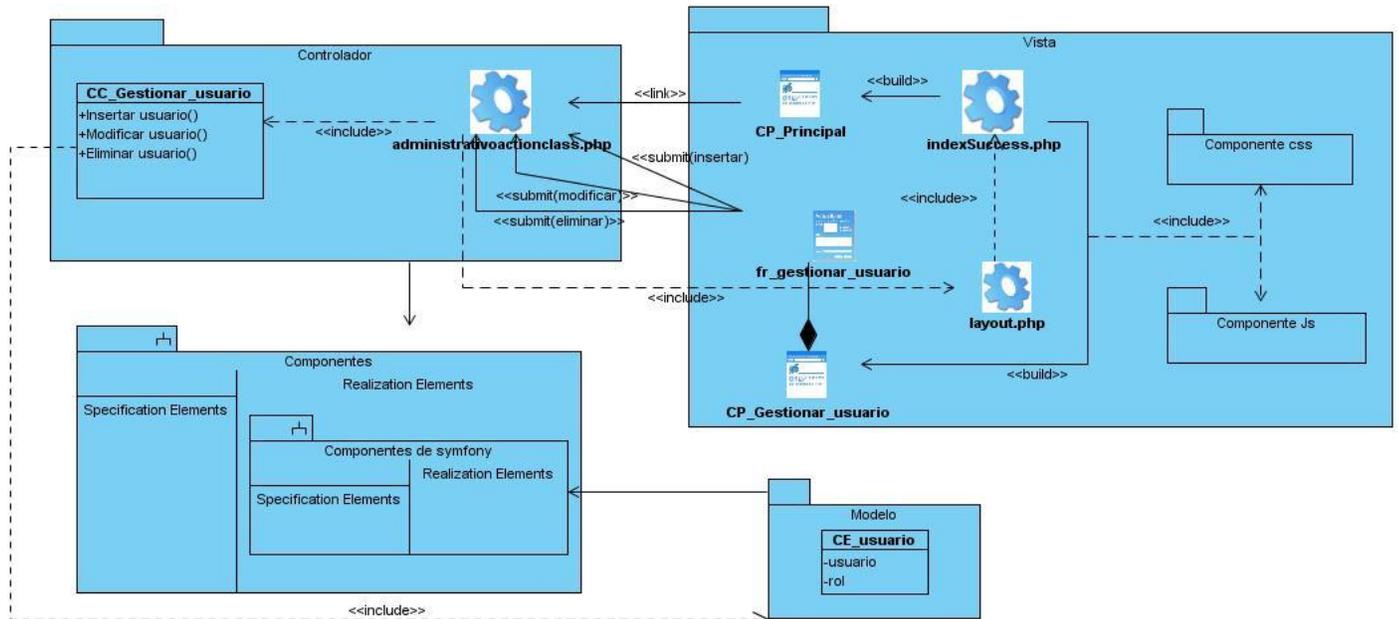
3.3.3 Diagrama de clases del diseño

En el diagrama de clases de diseño se especifican los atributos y métodos de cada clase. Son utilizados para modelar la vista de diseño estática de un sistema, esto incluye modelar el vocabulario del

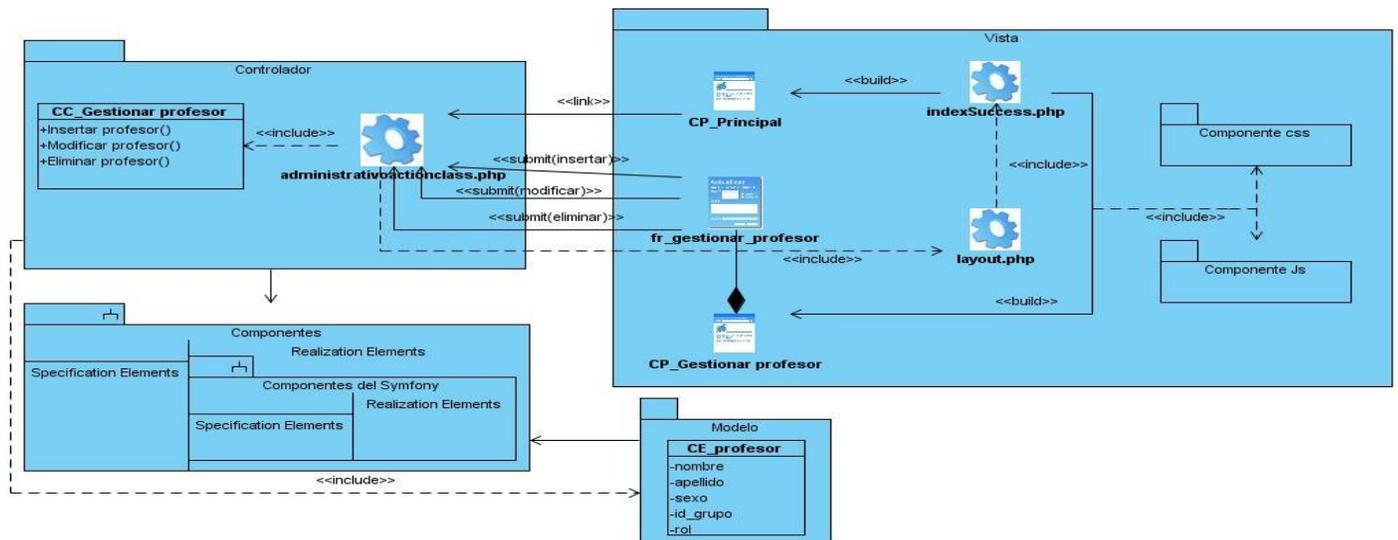
sistema, modelar esquemas o modelar las colaboraciones. Conforman la base para los diagramas relacionados: los diagramas de componentes y los diagramas de despliegue. Estos son importantes no sólo para visualizar, especificar y documentar modelos estructurales, sino también para construir sistemas ejecutables, aplicando ingeniería directa e inversa.

A continuación se muestran los diagramas de clases del diseño con estereotipos web de los casos de uso del sistema.

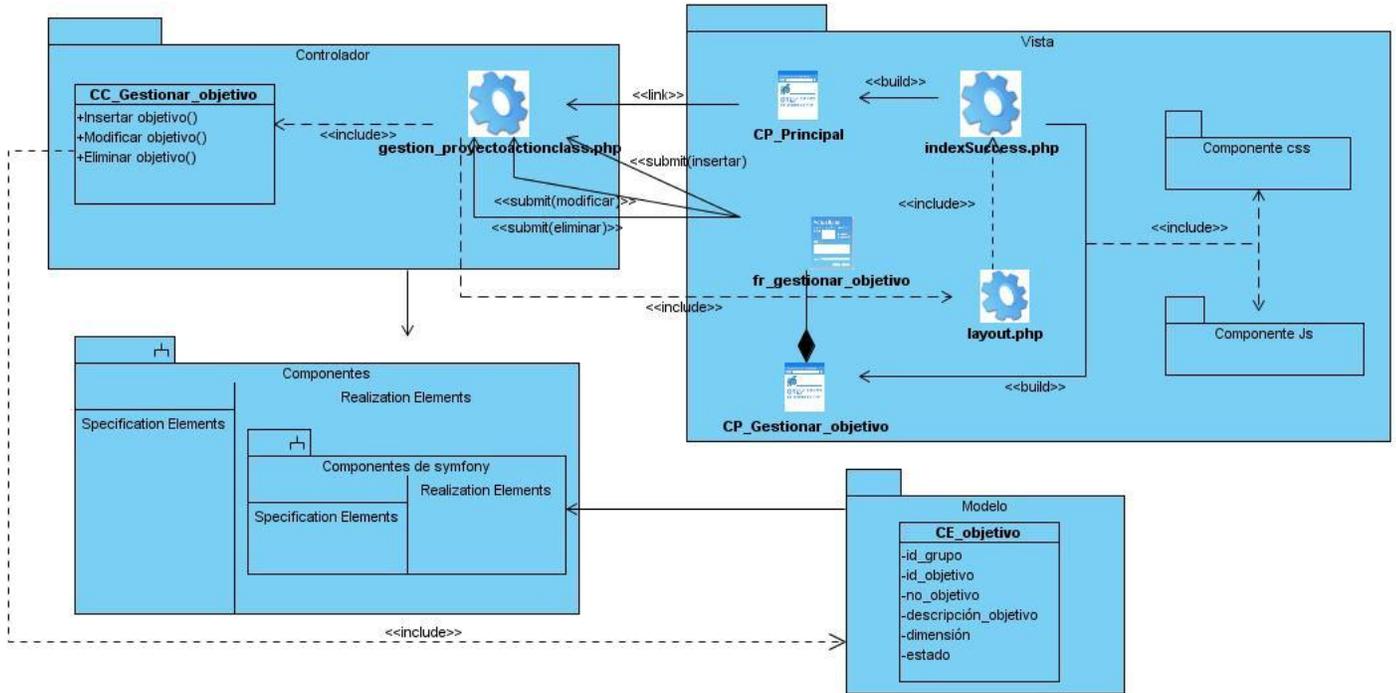
Diagramas de clases del diseño con estereotipos web del caso de uso Gestionar usuario



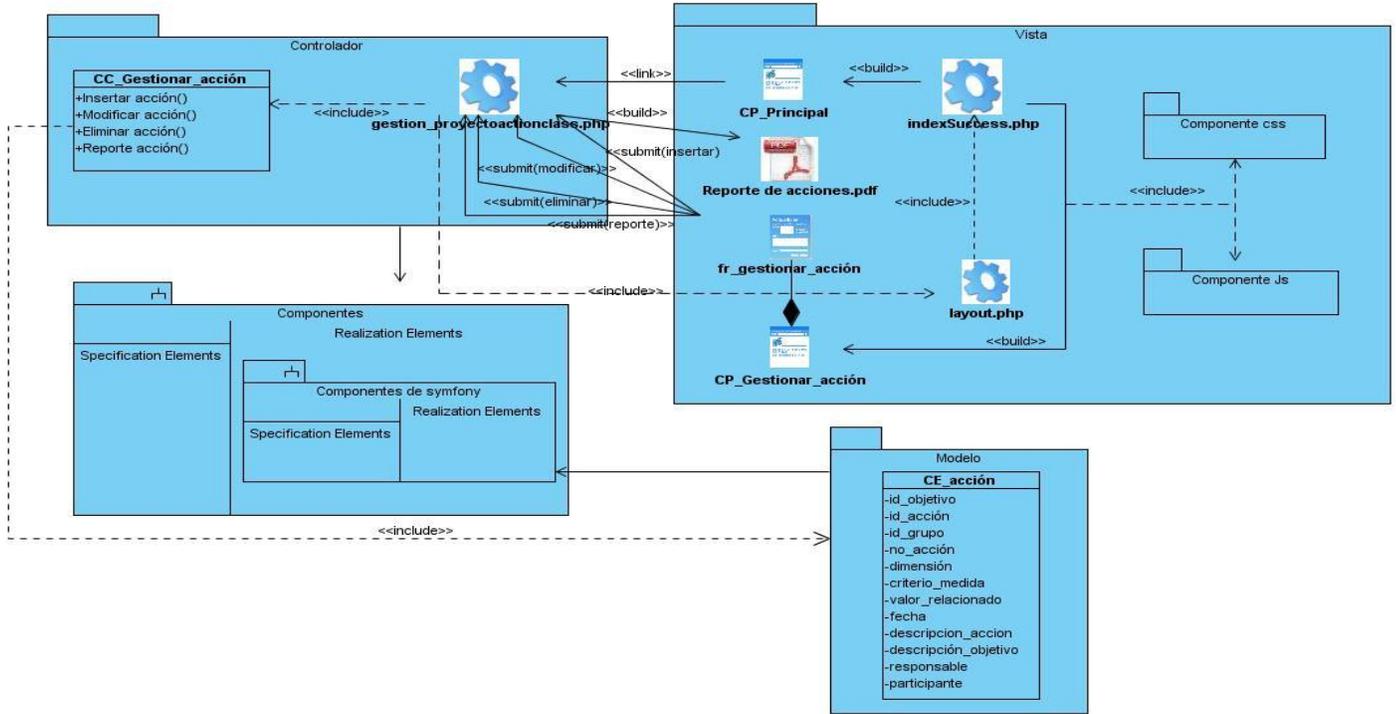
Diagramas de clases del diseño con estereotipos web del caso de uso Gestionar profesor



Diagramas de clases del diseño con estereotipos web del caso de uso Gestionar objetivo

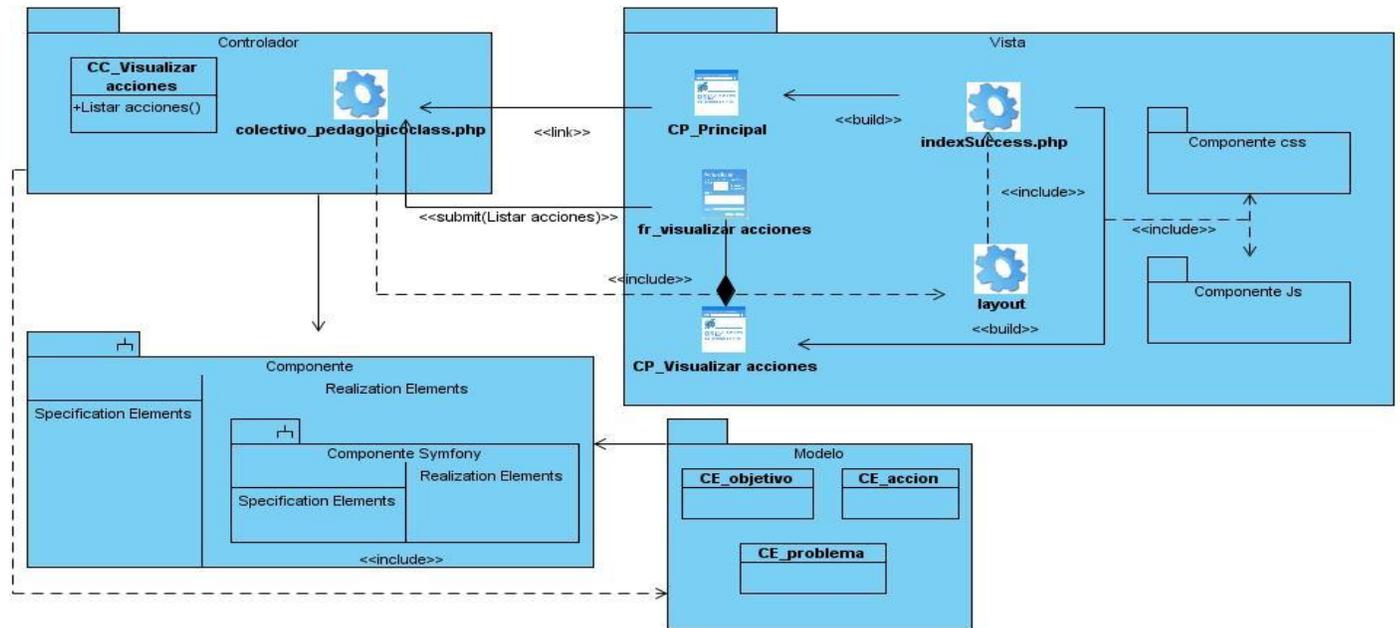


Diagramas de clases del diseño con estereotipos web del caso de uso Gestionar acción



Diagramas de clases del diseño con estereotipos web para los casos de uso Visualizar acciones.

Nota: En el diagrama presentado a continuación, visualizar acciones es como se representa en un diagrama de diseño: Visualizar problema, Visualizar objetivo ó Visualizar acción, pues es un proceso común utilizando cada uno su modelo correspondiente.

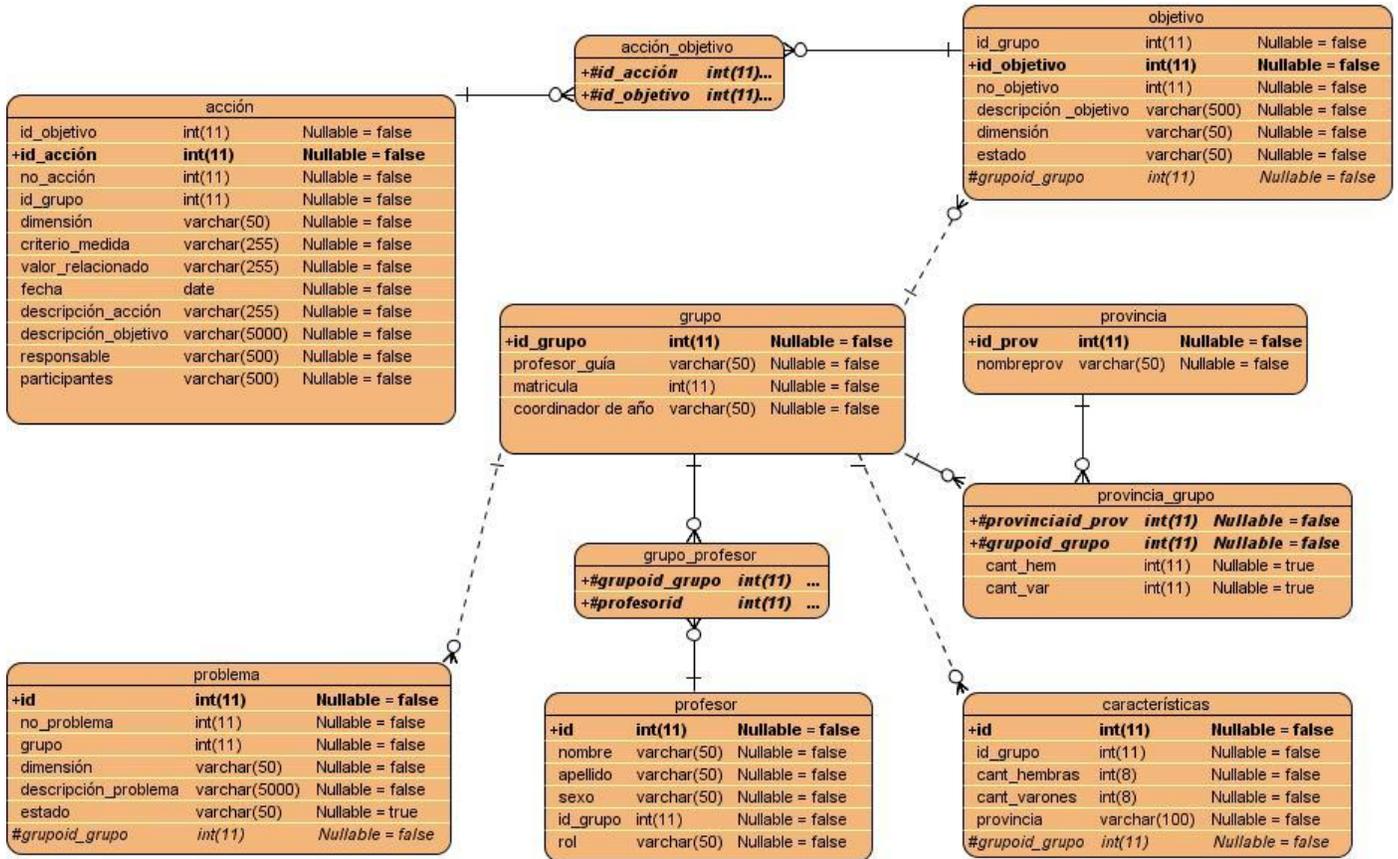


3.4 Diseño de base de datos

Los sistemas web en la actualidad son desarrollados utilizando tecnologías cliente-servidor. En la mayoría de los casos cuentan con bases de datos robustas que son las encargadas de almacenar todos los datos que se muestran. Para el sistema web gestión educativa, se ha desarrollado una base de datos que almacene toda la información que se controla en la misma.

3.5.1 Modelo de datos

El modelo de datos describe la representación lógica de las clases persistentes, previendo que información del sistema será soportada por una base de datos relacional.



3.5.2 Descripción de las tablas

Las principales entidades de la base de datos son descritas a continuación.

Nombre: problema		
Descripción: Esta tabla almacena los problemas detectados en el grupo y que son registrados en la aplicación.		
Atributo	Tipo	Descripción
Id	int(11)	Se almacena un id que se genera automático.
no_problema	int(11)	Almacena el número de la acción según se registre en la aplicación.
grupo	Int(11)	Almacena el grupo al que pertenece el problema registrado.
dimensión	varchar(50)	Se almacena la dimensión a la que

		pertenece el problema.
descripción_problema	Varchar(5000)	Se almacena la descripción correspondiente al problema.
estado	Varchar(50)	Almacena el estado de solución que presenta el problema.

Nombre: objetivo		
Descripción: Esta tabla almacena los objetivos del proyecto educativo en el grupo.		
Atributo	Tipo	Descripción
Id	int(11)	Se almacena un id que se genera automático.
id_grupo	int(11)	Se almacena el grupo al que pertenece el objetivo.
Id_objetivo	Int(11)	
no_objetivo	int(11)	Almacena el número del objetivo según se registre en la aplicación.
descripción_objetivo	varchar(5000)	Se almacena la descripción correspondiente al objetivo.
dimensión	varchar(50)	Se almacena la dimensión a la que pertenece el objetivo.
estado	varchar(50)	Almacena el estado de cumplimiento del objetivo.

Nombre: acción		
Descripción: Esta tabla almacena los datos de las acciones que se registran en los proyectos educativos.		
Atributo	Tipo	Descripción
Id	int(11)	Se almacena un id que se genera automático.
id_objetivo	int(11)	Almacena el id del objetivo al que pertenece.

Id_acción	int(11)	Almacena el id de la acción.
no_acción	int(11)	Almacena el número de la acción según se registre.
Id_grupo	int(11)	Almacena el id del grupo al que pertenece.
dimensión	varchar(50)	Se almacena la dimensión a la que pertenece la acción.
criterio_medida	varchar(255)	Almacena el criterio de medida de la acción.
valor_relacionado	varchar(255)	Almacena el valor relacionado de la acción.
fecha	date	Se almacena la fecha de cumplimiento de la acción.
descripción_acción	varchar(5000)	Se almacena la descripción correspondiente a la acción.
descripción_objetivo	varchar(5000)	Se almacena el objetivo al que corresponde.
responsable	varchar(500)	Almacena la persona responsable de que se cumpla la acción propuesta.
participante	varchar(500)	Almacena las personas que estarán involucradas en el cumplimiento de la acción.

Nombre: grupo		
Descripción: Esta tabla almacena los datos de los grupos que se gestionan en los proyectos educativos.		
Atributo	Tipo	Descripción
Id	int(11)	Se almacena un id que se genera automático.
id_grupo	Int(11)	Almacena el id del grupo que se registra en la aplicación.
profesor_guía	varchar(50)	Almacena el profesor encargado de realizar en proyecto educativo del grupo.
matrícula	int(11)	Almacena la cantidad de estudiantes del grupo.
coordinador_año	varchar(50)	Almacena el coordinador de año que le

		corresponde al grupo.
--	--	-----------------------

Nombre: profesor		
Descripción: Esta tabla almacena los datos de los profesores que interactúan en los proyectos educativos.		
Atributo	Tipo	Descripción
Id	int(11)	Se almacena un id que se genera automático.
nombre	varchar(50)	Almacena el nombre del profesor.
apellido	varchar(50)	Almacena el apellido del profesor.
sexo	varchar(50)	Almacena el sexo del profesor.
Id_grupo	int(11)	Almacena el id_grupo que atiende el profesor.
rol	varchar(50)	Almacena el nombre del profesor.

Nombre: características		
Descripción: Esta tabla almacena las características de los grupos que gestionan los proyectos educativos.		
Atributo	Tipo	Descripción
Id	int(11)	Se almacena un id que se genera automático.
Id_grupo	int(11)	Almacena el id_grupo que presenta la característica.
cant_hembras	int(8)	Almacena la cantidad de hembras que existen en el grupo.
cant_varones	int(8)	Almacena la cantidad de varones que existen en el grupo.

provincia	varchar(100)	Almacena la cantidad de estudiantes por provincia.
------------------	--------------	--

Conclusiones

En este capítulo se estudió los temas relacionados con la arquitectura seleccionada para el desarrollo del sistema, se siguió paso a paso la documentación de la metodología utilizada, en la creación del mismo. Está representada por el análisis y el diseño del sistema con sus correspondientes modelos, además del modelo de base de datos con los patrones de diseño que se utilizaron.

Capítulo 4: Implementación

Introducción

Durante la etapa de diseño se refinan las clases con el objetivo de dar paso a la implementación del sistema. También se toman las medidas pertinentes para el diseño de un sistema consistente con el entorno en el que se implementará.

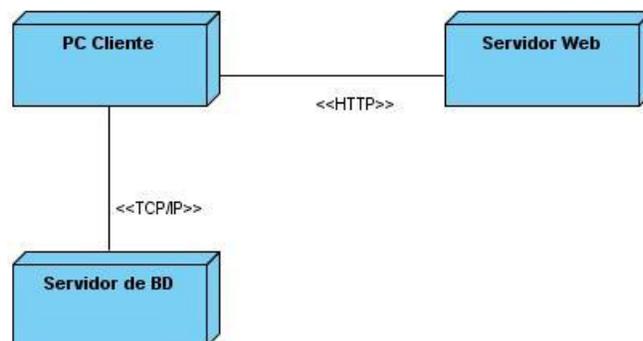
De una correcta implementación, que dé solución a los requisitos planteados, depende en gran medida que de las pruebas que se realizarán una vez implementado el sistema, se obtengan los resultados esperados por los desarrolladores.

4.1 Implementación

La implementación es el flujo de trabajo en que se desarrolla el sistema en términos de componentes: ejecutables, ficheros de código fuente, scripts, entre otros. Tiene como objetivo principal desarrollar la arquitectura y el sistema como un todo, así como definir la organización del código [32].

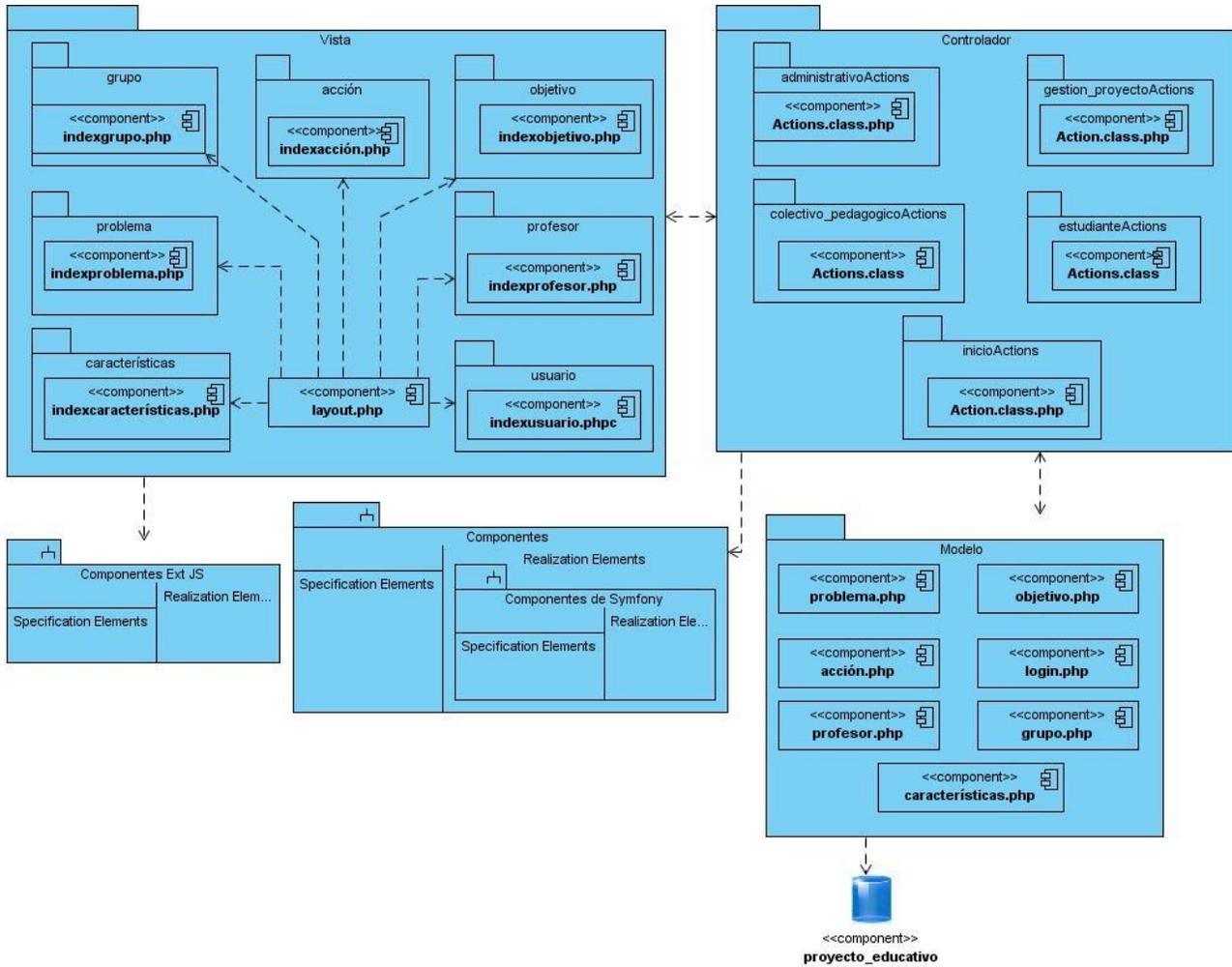
4.1.1 Modelo de despliegue

El Modelo de despliegue es un modelo de objetos compuesto por nodos y sus relaciones, que describen la distribución física de un sistema.



4.2 Diagrama de componentes

El diagrama de componentes muestra las dependencias lógicas y las organizaciones de componentes de software. Estos pueden utilizar paquetes para agrupar elementos del modelo. Han sido agrupados igualmente en paquetes para facilitar su comprensión y poder detallar mejor estos. Los paquetes que se mostrarán le dan seguimiento a los mostrados durante el diseño.



Conclusiones.

En el presente capítulo durante la implementación se cumplieron los principios de diseño establecidos, así como los estándares de interfaz e implementación determinados. Se presentaron los diagramas de componentes y de despliegue, donde fue descrita la distribución física del sistema y sus componentes.

Conclusiones

- ❖ La solución informática existente para gestionar la información educativa, no responde a las necesidades que se presentan para ser utilizado en la gestión de los proyectos educativos de la facultad 7.
- ❖ Las herramientas y tecnologías utilizadas para el desarrollo del sistema entre las que se encuentra, la metodología RUP, el lenguaje de modelado UML, la herramienta de modelado Visual Paradigm, el lenguaje de programación PHP, el framework de desarrollo Symfony, el gestor de base de datos MySQL junto a la herramienta de desarrollo NetBeans, constituyeron las adecuadas para su implementación.
- ❖ En la realización del diseño del sistema se utilizó la librerías Ext JS, con la cuál se logró una interfaz mas amigable y sencilla, además de utilizar el patrón de modelo vista controlador, que permitió la organización del sistema.
- ❖ Se obtuvieron todos los artefactos que propone la metodología de desarrollo RUP, entre los que se encuentran diagrama de casos de usos del negocio, del sistema, diagrama de componentes, diagrama de despliegue, entre otros.
- ❖ Se obtuvo el sistema GEDUC, capaz de gestionar los proyectos educativos de la facultad 7 de la UCI.

Recomendaciones

Por la experiencia adquirida la autora recomienda:

- ❖ Adaptar el sistema para su utilización en otras facultades de la universidad.
- ❖ Incorporarles nuevas funcionalidades que puedan elevar más, el resultado de los proyectos educativos a nivel de universidad.

Referencias Bibliográficas

1. _____, El Proyecto educativo. 2007. Disponible en:
<http://lascompetenciasbasicas.files.wordpress.com/2008/02/proyectoeducativo.pdf>
2. _____, El Proyecto Educativo. Una mirada desde las Universidades Cubanas.
<http://www.eumed.net/rev/ced/04/gcp.htm>
3. Igual a referencia 1.
4. _____, Gestión Estratégica del Proyecto Educativo. 1997.
5. Polo, Juan Diego, Enter the Group-Para gestionar proyectos educativos. 2010. Disponible en: <http://www.whatsnew.com/2010/10/27/enter-the-group-para-gestionar-proyectos-educativos/>
6. Igual a referencia 6.
7. Peña González, Amarilis Antonia, Tesis: Herramienta para automatizar el proceso de evaluación del programa de mejoras en los proyectos productivos de la UCI, basados en el modelo CMMI. 2010.
8. Igual a referencia 7.
9. Joskowicz, Ing. José, Reglas y Prácticas en eXtreme Programming. 2008.
10. Juan Palacio, El Modelo Scrum. 2006.
11. Igual a referencia 10.
12. METODOLOGÍA RUP [En línea] [Citado el: 8 de Febrero de 2009.]
<http://teleformacion.uci.cu/mod/resource/view.php?id=2241>.
13. González Cornejo, José Enrique, ¿Qué es UML? Disponible en
<http://www.docirs.cl/uml.htm>.
14. Hernández Orallo, Enrique. El Lenguaje Unificado de Modelado (UML) Disponible en
<http://www.disca.upv.es/enheror/pdf/ActaUML.PDF>.
15. Igual a referencia 7.
16. _____, Modelo Vista Controlador Disponible en <http://www.nelste.com/modelo-vista-controlador/>
17. Fahnle, Pablo. ASP: NET [En línea] [Citado el: 4 de Febrero de 2009.]
http://www.programacion.com/asp/articulo/aspnet_quees/.
18. PHP.net. Php.net. *Php.net*. [En línea] 2009. [Citado el: 15 de Enero de 2010.]
<http://php.net>.
19. Alvarez, Miguel Ange. JAVA [En línea] [Citado el: 5 de Febrero de 2009.]
<http://www.desarrolloweb.com/articulos/497.php>.

20. Delgado, Lester. Tesis: Herramienta para la revisión y seguimiento de errores de la documentación generada en los Proyectos de la Facultad 7.Junio 2009.
21. Igual a referencia 7
22. Visual Paradigm. Visual-Paradigm.com. Visual-Paradigm.com. [En línea] 2009. [Citado el: 10 de Enero de 2010.] <http://www.visual-paradigm.com/>.
23. Netbeans [Online] [Cited: noviembre 23, 2010.]
http://www.dadisokd.net.au.net/index.php/component/content/article/44-software-libre-dadisokd/94-netbeans-dadisokd_
24. ZEND STUDIO [En línea] [Citado el: 16de Febrero de 2009.]
http://www.ZendStudio.org//Zend_Studio
25. Igual a referencia 20
26. PostgreSQL Global Development Group. Postgresql.org . Postgresql.org. [En línea] 2009. [Citado el: 16 de Enero de 2010.] <http://www.postgresql.org/>.
27. _____ Conferencia Ingeniería de Software Flujos de Trabajo de análisis y diseño.2009
28. Igual a referencia 27.
29. Igual a referencia 27.
30. _____ Arquitectura del Software. Capítulo 2.MVC.
31. Igual a referencia 7.
32. _____ Conferencia Ingeniería de Software Implementación 2009.
33. _____ Conferencia Ingeniería de Software Pruebas 2010.

Bibliografía

- ❖ _____, El Proyecto educativo. 2007. Disponible en:
<http://lascompetenciasbasicas.files.wordpress.com/2008/02/proyectoeducativo.pdf>
- ❖ _____, El Proyecto Educativo. Una mirada desde las Universidades Cubanas.
<http://www.eumed.net/rev/ced/04/gcp.htm>
- ❖ _____, Gestión Estratégica del Proyecto Educativo.1997.
- ❖ Polo, Juan Diego, Enter the Group-Para gestionar proyectos educativos.2010.Disponible en:
<http://www.whatsnew.com/2010/10/27/enter-the-group-para-gestionar-proyectos-educativos/>
- ❖ Amarilis Antonia, Tesis, Herramienta para automatizar el proceso de evaluación del programa de mejoras en los proyectos productivos de la UCI, basados en el modelo CMMI.2010.
- ❖ Joskowicz, Ing.José, Reglas y Prácticas en eXtreme Programming.2008.
- ❖ Palacio, Juan, El Modelo Scrum.2006.
- ❖ METODOLOGÍA RUP [En línea] [Citado el: 8 de Febrero de 2009.]
<http://teleformacion.uci.cu/mod/resource/view.php?id=2241>.
- ❖ José Enrique González Cornejo. ¿Qué es UML? Disponible en:
<http://www.docirs.cl/uml.htm>.
- ❖ Enrique Hernández Orallo. El Lenguaje Unificado de Modelado (UML) Disponible en:
<http://www.disca.upv.es/enheror/pdf/ActaUML.PDF>.
- ❖ _____, Modelo Vista Controlador Disponible en:
<http://www.neleste.com/modelo-vista-controlador/>
- ❖ Fahnle, Pablo.ASP: NET [En línea] [Citado el: 4 de Febrero de 2009.]
http://www.programacion.com/asp/articulo/aspnet_ques/.
- ❖ PHP.net. Php.net. Php.net. [En línea] 2009. [Citado el: 15 de Enero de 2010.]
<http://php.net>.
- ❖ Alvarez, Miguel Angel. JAVA [En línea] [Citado el: 5 de Febrero de 2009.]
<http://www.desarrolloweb.com/articulos/497.php>.

- ❖ Lesther Delgado. Herramienta para la revisión y seguimiento de errores de la documentación generada en los Proyectos de la Facultad 7.Junio 2009.
- ❖ Visual Paradigm. Visual-Paradigm.com. Visual-Paradigm.com. [En línea] 2009. [Citado el: 10 de Enero de 2010.]
<http://www.visual-paradigm.com/>.
- ❖ Netbeans [Online] [Cited: noviembre 23, 2010.]
<http://www.dadisokd.net.au.net/index.php/component/content/article/44-software-libre-dadisokd/94-netbeans-dadisokd>.
- ❖ ZEND STUDIO [En línea] [Citado el: 16de Febrero de 2009.]
http://www.ZendStudio.org//Zend_Studio
- ❖ PostgreSQL Global Development Group. Postgresql.org . Postgresql.org. [En línea] 2009. [Citado el: 16 de Enero de 2010.]
<http://www.postgresql.org/>.
- ❖ _____Conferencia Ingeniería de Software Flujos de Trabajo de análisis y diseño.2009
- ❖ _____Arquitectura del Software. Capítulo 2.MVC.
- ❖ _____Conferencia Ingeniería de Software Implementación 2009.
- ❖ 33. _____Conferencia Ingeniería de Software Pruebas 2010.
- ❖ 34. Alvarado Oyarce, Otoniel. GESTIÓN DE PROYECTOS EDUCATIVOS.
http://sisbib.unmsm.edu.pe/BibVirtual/libros/Educaci%F3n/gestion_proyectos/contenido.htm
- ❖ García Moreno Wilker. Proyectos Educativos Para Ferias de Ciencias.
- ❖ 36.Álvarez, Julio César. PROYECTO EDUCATIVO INSTITUCIONAL. 1997
[http://www.bnm.me.gov.ar/cgi-in/wxis.exe/opac/?IsisScript=opac/opac.xis&dbn=BINAM&src=link&tb=tem&query=PROYECTO%20EDUCATIVO%20INSTITUCIONAL&cantidad=&formato=&sala=.](http://www.bnm.me.gov.ar/cgi-in/wxis.exe/opac/?IsisScript=opac/opac.xis&dbn=BINAM&src=link&tb=tem&query=PROYECTO%20EDUCATIVO%20INSTITUCIONAL&cantidad=&formato=&sala=)